

login:

May/June 1992

Vol. 17, No. 3

From The Editor's Desk

Greetings from sunny Colorado Springs!

USENIX is certainly a busy organization this year. By the time you receive this issue, the Summer Conference in San Antonio will have occurred. I have seen the proceedings -- lots of good stuff. We'll have a report on the conference in a future issue.

The MicroKernel workshop had 340 attendees; it appears that MicroKernels are this year's hot topic. The Filesystems workshop proceedings have a great set of papers on all the latest production and experimental filesystem technology (including global filesystems, filesystem internals, object-oriented filesystems, high performance filesystems, log-structured filesystems, recovery protocols, replicated file services, specialized file-systems, massively scaled filesystems, NFS, AFS, and WORM filesystems). Neat stuff (but I like filesystems).

LISA VI (it's not just for large installations any more) will sport not only two days of tutorials but also a terminal room, vendor displays, BOFs, and parallel tracks. Wow. (I feel like Drew Kaplan as I write this). Submit your abstract for the technical sessions soon (see the Call in this issue)!

Some of you may never have attended a USENIX workshop. They are different from the big conferences in their 'flavor'. They are far more concentrated and tend to bring a focus to their topic that is hard to find at the main conferences (as counterpoint, their diversity is limited). If you need to come up to speed on an issue (e.g., filesystems, system administration, security, C++), it's hard to imagine a better, more cost-effective way (in terms of both time and money).

This newsletter improves (I hope!) with each issue. I'm particularly interested in articles from attendees at conferences and workshops that summarize their trip (and/or the proceedings). If you would like to contribute to future issues, please contact me at kosltad@bsd1.com.

See you at a future USENIX event!

RK

Features:

What's Out There?	3
<i>Jeff Kellein</i>	
Trip Report on NeXTWORLD	6
<i>Barbara J. Dyker</i>	
BSDI BOF Report.....	9
<i>Kevin Smallwood</i>	

Association News:

Election Results.....	10
Annual Report to Membership	11
Board Meeting Summary	13

Features (cont.)

Prime Time Freeware CD	14
Report from EurOpen.....	15
<i>Alain Williams</i>	

Eclectica:

Reflections	16
<i>Robert D. Carlitz</i>	

Can UNIX Designers Learn Anything from PCs?	17
<i>Jeff Haemer</i>	

An Update of UNIX-Related Standards Activities	19
<i>Stephen Walli</i>	

Book Reviews:

The Bookworm	28
<i>Peter Salus</i>	
Practical UNIX Security.....	29
<i>George Leach</i>	

Calendar of UNIX-Related Events	31
---------------------------------------	----

Announcements:

Winter 1993 Conference.....	32
LISA VI Conference	34
C++ Conference	35
Security Symposium	38

Publications	39
--------------------	----

Local User Groups	42
-------------------------	----

The closing date for submissions to the next issue of *login:* is June 24, 1992.



The UNIX and Advanced Computing Systems
Professional & Technical Association

General Information

login: is the official newsletter of the USENIX Association.

login: (ISSN 1044-6397) Volume 17, Number 3 (May/June 1992) is published bimonthly by the USENIX Association, 2560 Ninth Street, Suite 215, Berkeley, CA 94710. \$24 of each member's annual dues is for an annual subscription to *login*. Subscriptions for non-members are \$50 per year. Application to Mail at Second Class Postage Rates is Pending at Berkeley, California and additional offices. POSTMASTER: Send address changes to *login*, USENIX Association, 2560 Ninth Street, Suite 215, Berkeley, CA 94710.

Contributions Solicited

You are encouraged to contribute articles, book reviews and announcements to *login*. Send them via email to login@usenix.org or through the postal system to the Association office. The Association reserves the right to edit submitted material. Any reproduction of this newsletter in its entirety or in part requires the permission of the Association and the author(s).

Editorial Staff

Rob Kolstad, Editor <kolstad@usenix.org>
Ellie Young, Editor <ellie@usenix.org>
Carolyn S. Carr, Managing Editor and Typesetter <carolyn@usenix.org>
Michelle Dominijanni, Copy Editor

Membership and Publications

USENIX Association
2560 Ninth Street, Suite 215
Berkeley, CA 94710
Telephone: 510/528-8649
FAX: 510/548-5738
Email: <office@usenix.org>

UUNET Subscriptions

UUNET Technologies, Inc.
3110 Fairview Park Drive, Suite 570
Falls Church, VA 22042
703/ 204-8000
FAX 703/204-8001
<uunet-request@uunet.uu.net>

Conferences, Workshops & Symposia

Judith F. DesHarnais, Conference Coordinator
USENIX Conference Office
22672 Lambert Street, Suite 613
El Toro, CA 92630
Telephone: 714/588-8649
FAX: 714/588-9706
<conference@usenix.org>

Tutorials

Daniel V. Klein, Tutorial Coordinator
Telephone: 412/421-2332
<dvk@usenix.org>

Supporting Members

Digital Equipment Corporation
Frame Technology, Inc.
Matsushita Graphic Communication Systems Inc
mt Xinu
Open Software Foundation
Quality Micro Systems
Rational Corporation
Sun Microsystems, Inc.
Sybase, Inc.
UNIX System Laboratories, Inc.
UUNET Technologies, Inc.

USENIX Association Board of Directors

President:
M. Kirk McKusick <mckusick@usenix.org>
Vice President:
Michael D. O'Dell <mo@usenix.org>
Secretary:
Rob Kolstad <kolstad@usenix.org>
Treasurer:
Sharon Murrel <eowyn@usenix.org>
Directors:
Rick Adams <rick@usenix.org>
Ed Gould <ed@usenix.org>
Evi Nemeth <evi@usenix.org>
Barry Shein <bzs@usenix.org>

USENIX Association

Executive Director
Ellie Young <ellie@usenix.org>

Copyright © 1992 USENIX Association. The Association acknowledges all trade references made herein.

login: is printed on recycled paper.



What's Out There?

Jeff Kellem, Beyond Dreams
<composer@Beyond.Dreams.ORG>
Vol. 1, Issue 1

Intros... Maestro?

When acting as the writer for this column, "What's Out There?," I'll be writing about... well, what's out there. In this case, "there" is the net at large. Topics are expected to range from freely available software and documentation to methods of finding information to interesting discussions to whatever I feel like writing about.

I'm Jeff Kellem, the founder of Beyond Dreams, an organization which promotes information exchange and communication. This column is just one of the things that the organization is attempting to provide. If you'd like more information or to contact me, I can be reached via e-mail as <composer@Beyond.Dreams.ORG>.

The Overview

In this issue of "What's Out There?," you'll read about methods of finding software packages on the net, searching servers of information, where to get information on NREN, NSFnet policies, ANSnet policies, along with brief descriptions of neat information servers.

Where's that Software Package?

So, you've heard about this terrific piece of freely available software on the net, called the Pizza-Dude automatic ordering software. But, where can you get it? You're about to find out.

There's a service available at various places on the net (around the world), called archie. Archie is a database of anonymous ftp sites' directory listings. It was developed by Peter Deutsch, Alan Emtage, and Bill Heelan of McGill University. There are currently (at least) eight archie servers available on the net:

archie.mcgill.ca (Canada)
archie.sura.net (USA, MD)
archie.ans.net (USA, NY)
archie.funet.fi (Finland/Mainland Europe)
archie.rutgers.edu (USA, NJ)
archie.au (Australia)
archie.doc.ic.ac.uk (Great Britain, Ireland)
archie.unl.edu (USA, NE)

There are several ways to query an archie server. You can:

telnet to the archie server, logging in as archie (no password is required),
e-mail a message to *archie@archie.mcgill.ca* (for example), or
use one of the archie clients.

When you telnet to an archie server, you're presented with the prompt:

```
archie>
```

At the "archie>" prompt, you can enter "help" to find out how to use the "prog" command to do searches, along with everything else you can currently do with archie. To exit archie, type "quit".

To find out how to use the e-mail interface, send e-mail to *archie@archie.mcgill.ca* with a body consisting of the word "help".

There are two primary archie clients available, archie (from the Prospero¹ distribution) and xarchie (and X Windows client). Both should be available via anonymous ftp from most (if not all) of the archie servers. You can also find the archie clients via anonymous ftp from *ftp.cs.widener.edu* in the /pub/archie directory.

Below is some sample output of the standalone archie client via:

```
archie perl-4.019.tar.Z
```

to search for version 4.019 of the Perl distribution. Output from the 'prog perl-4.019.tar.Z' command by telnet-ing to an archie server is similar.

```
Host aeneas.mit.edu
```

```
Location: /pub/gnu  
FILE -rw-r--r-- 801616 Nov 13 16:03 perl-4.019.tar.Z
```

```
Host ftp.uu.net
```

```
Location: /packages/gnu  
FILE -rwxr-xr-x 801616 Nov 13 11:03 perl-4.019.tar.Z
```

```
Host wuarchive.wustl.edu
```

```
Location: /mirrors2/gnu  
FILE -rw-rw-r-- 801616 Nov 13 10:03 perl-4.019.tar.Z
```

1. The complete Prospero distribution is available via anonymous ftp from *cs.washington.edu* in /pub/prospero.tar.Z. Prospero will be described in a future issue.

What if I don't know the name of a package?

Archie can also help here. A "whatis" database is part of archie, providing descriptions to over 3,500 freely available "software packages and informational documents located on the Internet."² The contents of the whatis database is expected to be expanded in coming months to include such items as names and locations of online library catalogues, archive sites for mailing lists and USENET newsgroups, compilations of Frequently Asked Questions lists, among others.

Another Method -- Charlie

There's also another database of freely available packages on the net, called charlie, developed by Mike Stump <mrs@charlie.secs.csun.edu>. Charlie is currently a user-supported database which includes descriptions of software packages and where to find them. Mike considers this a first step to another goal, to make charlie a software information server that "knows all about software" -- where to get it, what it requires to build, etc. Ideally, you would be able to run some client that would grab a copy of some software package and anything that it might require, configure and build each of them, and install it. To try out charlie, in its current form:

```
telnet charlie.secs.csun.edu 5742
```

Other Places to Look for Packages

Other places you can look or keep an eye on to locate neat software packages are the USENET newsgroup comp.archives, the various comp.-sources.all newsgroups, newsgroups and mailing lists pertaining to the type of package you may be looking for, and the comp.archives WAIS server.

So, What's WAIS?

WAIS (pronounced "ways") stands for Wide Area Information Server. It's a set of programs (or, more specifically, a protocol) that allows users to search and access different types of information from a single interface. This information can be practically anything, from text to sound to images to whatever you can think up. The information can reside anywhere and on many different computer systems. The WAIS protocol is an extension of the ANSI Z39.50-1988 information retrieval protocol. It will be revamped in the future to be based on the newer Z39.50-1991 protocol.

There are several interfaces to WAIS currently available, with more in development, including a GNU Emacs, a dumb terminal, a shell, a Mac, a NeXT, and an X Windows interface. Most of these, including the server code so that you can startup your own WAIS server, are freely available via anonymous ftp from think.com in the /wais directory.

Once you start up a WAIS client, you specify what's called a source to search upon.³ You can ask multiple sources for information. Then, you ask the source(s) a question. A question consists of a phrase. With the current sample server/client implementations, this phrase is basically considered a set of keywords to search for, based on weights/percentages of each word in the documents (you're searching upon).

But, since WAIS really just specifies the protocol for the client and server to use for communication, the underlying search on the server could just as well use various natural language queries upon its information. With the WAIS protocol, it is also possible to ask the source(s) for other documents which are similar to the ones found; this is called relevance feedback.

To try out the dumb terminal interface, without having to compile some client, telnet to quake.think.com and login as user "wais".

When you login as "wais", a list of WAIS sources will be displayed. Help for each screen is available via the '?' key. Say, for example, we want to search the "bible" source for the keywords "flood" and "noah". Here's what you would do:

In the Source Selection screen, type
"/bible<RETURN>" (whether <RETURN> means pressing the RETURN or ENTER key); this will move the cursor to the bible source.

Press the <SPACEBAR> to select the currently highlighted (or pointed to) source to search upon. You can search upon multiple sources, just by moving to them and selecting them with the <SPACEBAR>.

Press the 'w' key to enter keywords to search for; the Keywords: prompt will appear.

At the Keywords: prompt, enter "flood noah":
Keywords: flood noah

2. From "whatis.archie", available via anonymous ftp fromarchie.mcgill.ca in /archie/doc/whatis.archie.

3. A "source" specifies a server of information and how the client can contact it, along with some other information.

Press <RETURN> to execute the search.

A list of documents that include those keywords will be displayed in the Search Results screen.

To view a document, move to it and press the <SPACEBAR>.

There are other things you can do with those documents; press the '?' key for help on the Search Results screen.

To quit, press 'q'.

Hopefully, that will be enough to get you started with SWAIS, the Simple WAIS interface you just connected to. Enjoy.

NREN, ANSnet, NSFnet info

We leave the realm of locating software packages and information access methods to find information on specific topics of interest. Recently [around the beginning of the year], on the com-priv mailing list (COMmercialization and PRIVatization of the net), there has been some discussion and controversy regarding ANS's announcement regarding commercial traffic, the NSFnet, and other networks. The announcement will be left out to preserve space, but I'm here to point you to the information to read.

Items of interest regarding ANS via anonymous ftp from ftp.ans.net include:

Press Releases

```
/pub/info/press-releases/ans.creation  
/pub/info/press-releases/ans.co+re.creation  
ANS's plan and connect agreements  
/pub/info/ans-plan.txt  
/pub/info/midlevel-guide.txt  
/pub/info/connect-agreement.txt
```

You can obtain information regarding the NSFnet, along with the text of the High Performance Computing Act of 1991 (otherwise known as the NREN bill), via anonymous ftp from nis.nsf.net in the nsfnet directory. Three items of interest are:

```
nsfnet:nrenbill.txt  
nsfnet:nren.txt  
nsfnet:gorebill.1991-txt
```

To join the com-priv mailing list, send e-mail to *com-priv-request@uu.psi.com*. For discussion of the NREN, you can join the nren-discuss mailing list by sending e-mail to:
nren-discuss-request@uu.psi.com.

The Commercial Internet Exchange

While we're on the topic of commercialization and privatization of the net, you may also be interested to find out more information about CIX, the Commercial Internet Exchange Association, Inc. The CIX was founded in August, 1991 "to provide a non-restrictive packet interchange for TCP/IP and OSI traffic" between public data internetwork (PDI) service providers. The founding members are CERFnet, PSInet, and AlterNet. For more information on CIX, ftp to cix.org or contact:

The CIX Association
3110 Fairview Park Drive, Suite 590
Falls Church, VA 22042
Tel: +1 703-876-5050
Fax: +1 703-876-5059
e-mail: <info@cix.org>

Back to the Fun Stuff -- Weather & Geographic Servers

Now, back to the fun stuff. One thing I do practically every day is check on the weather. There are various weather servers on the net, of which I'll mention two that I use.

The Department of Atmospheric, Oceanic, & Space Sciences at the University of Michigan maintains a database of weather information, called the Weather Underground, for the entire United States. Some of the items available are current weather conditions and forecasts for cities around the US, ski conditions, and severe weather condition reports. To try it out:

```
telnet madlab.sprl.umich.edu 3000
```

They also have weather information for Canada, earthquakes, hurricane advisories, and long-range forecasts.

Another weather server I use is primarily for Boston, MA and surrounding areas, accessible via:

```
finger weather@synoptic.mit.edu
```

There are other weather servers across the net, but I'm running out of space and time to describe them here. Perhaps some other time.

Geographic Name Server

I'll squeeze in one more sometimes useful information server, the Geographic Name Server, maintained by Merit, Inc. This is a database of information on cities in the United States, along with some international locations. It provides an array of information including the area code, longitude and latitude, elevation, population, time

zone, and zip code. You query the database by giving it a city name, zip code, or regular expression. Some example queries are "Boston, MA", "02215". To use the server:

```
telnet martini.eecs.umich.edu 3000
```

Coming Attractions?

In future issues, you'll read more about Prospero,

gopher, WorldWideWeb, Hyperbole, and various other information server related items. There will also be information on various online text projects, such as Project Gutenberg and the Online Book Initiative (OBI). What else? Whatever comes to mind.

Till next time, I'll see you on the net.

Trip Report & Opinion

NeXTWORLD Expo

Barbara J. Dyker
<dyker@locutus.cs.colorado.edu>

Steve Jobs opened the first NeXTWORLD Expo in San Francisco held concurrently with Uniforum and USENIX, Jan 22-24. As most people know, the NeXT computer is a Mach UNIX based system. In the last year NeXT has been catching on like wild fire in specialized commercial and government environments while continuing to be popular in academia -- with sales and third party applications growth of over 400% in 1991. Although I'm a self proclaimed NeXT addict, I'll do my best to provide the highlights of the Expo without sounding like a sales pitch. This Expo was the platform for new announcements from NeXT.

The first NeXTWORLD Expo, on the tails of MacWORLD the week before, included a 2-day user conference, a 3-day developer conference, 2 days of vendor expositions, a full day "global" user group meeting, and some keynotes and promotional seminars. The attendance was rather impressive for a first event -- the morning of the keynote, registration was swamped: ~5000 total attendees, and 125 vendors. Every state in the US was represented as was every major continent in the world. The Expo was planned in a dizzying 88 days from concept to keynote. The various vendor specific WORLD expos used to be just for the PC and Mac crowd. I thought the NeXT WORLD Expo was fairly natural since NeXT straddles the PC and UNIX realms, misunderstood by the buzzword seekers at Uniforum. However, Sun Microsystems has one scheduled for March.

Jobs mentioned in recent published interviews and again in his keynote that NeXT has found their strength in their NeXTSTEP object oriented

system and development environment. NeXT customers are finding that custom application development with NeXTSTEP is accomplished at three to ten times the pace of other UNIX and GUI systems. This is the direction NeXT is now emphasizing. In most areas, NeXT has found their biggest competitor to be Sun. Jobs all but declared war on Sun. The conclusion of an independent programmer and developer survey by Booz, Allen & Hamilton, dated Jan 7, 1992, (where half the respondents were experienced in software development on Suns) revealed: "Over 82% of the developers and programmers surveyed ranked NeXTSTEP higher than other environments they had used (Sun, Macintosh, MS-DOS) in all major areas -- development environment completeness, application quality, application maintainability, and development time." Sounds great? This third turn in NeXT strategy has many academic folks feeling orphaned.

In keeping up with the Joneses, Jobs announced and demonstrated all the new toys including the NeXTstation Turbo: a 33MHz 68040 workstation rated at 25 MIPS performance. The 25MHz station is staying around as an entry level system. The turbo is available in the same grayscale and color monitor options as the station was. NeXTSTEP 3.0 was also rolled out. New features bundled in the 3.0 release include: full PostScript level 2 with full pantone color support, Pixar Renderman (finally), integrated protocol gatewaying support for Novell Netware and AppleShare, localization support for 6 languages, public key encryption built-in for confidential mail, a database object kit, and practically unrestricted object messaging. A long awaited color laser printer was also announced: a 360dpi CMYK plain paper printer that lists under \$4k! Like the original NeXT printer, the PostScript interpreter is on the NeXT, not in the printer, which keeps the cost (and the speed) down. Last, but certainly not least, is NeXTSTEP 3.0 for Intel 486 based PCs. The i486 version of NeXTSTEP 3.0

is the "same, same, same" as the NeXT version -- a complete port, not a different system.

Yeah, yeah. So another vendor has made announcements about new hardware and software, big deal. In attending USENIX, Uniforum, and the NeXTWORLD Expo, what I found most interesting was the similarity in technological direction. However, while the rest of the UNIX community is debating standards and trying to figure out the best way to back hack new technology into existing systems, NeXT is shipping features everyone else is only talking about or experimenting with, but without sacrificing compatibility and interoperability -- actually improving it, at least, where it fits their goals. For four or so years now NeXT has provided full WYSIWYG on screen with Display PostScript, multi-media mail and other applications, rich GUI object libraries and development tools, threads and distributed processing, and CD quality sound. Those are now givens, and NeXT is moving to more exciting territory. Yes, the NeXT runs X, but who cares?

The inclusion of Postscript level 2, full pantone colors, and Renderman provides true WYSIWYG color and 3D imaging with the NeXTDimension system and the new color printer. Renderman is incorporated into NeXTSTEP such that integrating 2D and 3D images is just a drag of the mouse.

Internetworking UNIX systems with MS-DOS PCs and Macintoshes has for a long time been a complicated, time consuming, and often costly endeavor. Public software solutions practically take a networking guru to figure out and commercial solutions are typically expensive. Most solutions don't solve the entire problem or do it in a not so usable manner. NeXTSTEP 3.0 provides Novell and AppleShare client support built into the system -- apparently in the NFS code. In that way, files on Novell or AppleShare can be accessed just like any other local or networked files on the NeXT. Even opening the files on the NeXT does the right thing -- no manual file conversions necessary. Printing to Novell or AppleTalk printers is also supported. Unfortunately you still need NFS to access NeXT files from a PC or Mac -- no server capabilities yet. I shouldn't forget to mention that ISDN support is also included in 3.0. Only a hardware interface to the line is required.

The Japanese market is excited over NeXT. NeXT is the only system that provides complete language support with Kanji under display postscript. Almost half of NeXT's business is overseas. The president of Canon spoke at the

keynote. He extolled NeXT for the Kanji version of 3.0 and offered no disparaging remarks. Kanji is supported in a separate release of NeXTSTEP. The full support of language localization allows the same application binary to work for any supported language without having to actually include language support in the application. Supported languages are: English, French, Spanish, German, Italian, and Swedish. The new ISDN support is also crucial for the users overseas.

There has been much work in recent years with multi-media mail and mail encryption. NeXT implemented their own public key encryption scheme called FEE, Fast Elliptic Encryption, and incorporated it into their multi-media Mail application. FEE uses a highly optimized elliptic-curve algebra methodology. FEE provides the public key and encrypts the private key. The data itself is encrypted with Digital Encryption Standard, DES.

Using FEE encrypted mail on the NeXT is simple. To encrypt mail for delivery, one merely selects a button. The message is encrypted using the recipient's public key. To read encrypted mail, the recipient needs to enter their own private key -- no one else can read their encrypted mail. NeXT has applied for a patent on FEE but plans to make the methodology publically available, if at all possible.

The database object kit for Interface Builder on the NeXT is another welcome addition to the palettes available with the system. To use it, one first needs to create an object that describes the specific database server structure to use and where it resides, locally or remotely on another system, presumably inheriting attributes from a prototype. To then build custom applications as query or edit interfaces is down right elementary: drag the interface objects from the database object kit palette into place and "connect" (alternate-drag) each to their target in the specific database object that describes the server. In demonstrating this, it took less than five minutes to develop and test a database interface application that performed complex and multi-media database queries.

The simple change of removing restrictions on object messaging opened up an incredibly powerful feature. In UNIX, many people have used hard or soft links in the filesystem to allow entire files to appear in more than one place at a time. Now the data in files can be included in other files with object linking (instead of pasting) with only a keystroke or mouse click difference. This provides a much more effective environment for collaborative work where pieces of the whole are

done by separate people. In a gross sense this accomplishes what an #include does, however you see the data, not the reference to it, and it works in any application that supports the standard cut/paste functions.

The announcement of the i486 port of NeXTSTEP 3.0 is quite significant. Many PCs in the commercial world are running some flavor of UNIX. The addition of Appleshare and Novell Netware support makes NeXTSTEP 3.0 a much less risky transition to the power of a UNIX OS for already mixed sites. The goal used to be a workstation on every desk. Now that there are often two to three workstations on a desk to accomplish different tasks, the goal is "unify the desktop," according to Jobs -- with a NeXT, of course.

The new features described here don't represent an exhaustive list of added functionality and changes. A few other features in the new 3.0 release include CD-ROM support, better sound handling, support for Mac 1.44mb floppies, built-in context-sensitive help, etc. The Spring issue of NeXTWORLD magazine provides more details. The CD-ROM support also closes the door on the OD. Not only will NeXT no longer sell the magneto optical drive, the new hardware doesn't even support it. Many optical drive owners are irate. It's more than likely a vendor will soon provide a SCSI interface for the useful but long obsolete OD. NeXT also dropped the 36 pin SIMMs and opted for 72 pin SIMMs for all models. The color station already used the 72 pin SIMMs. The NeXT Expo vendor exhibits were a welcome

change from the now tiring sales speak, trinket wars, and vaporware of Uniforum. Most of the booths were small and to the point: some with gimmicks and some with innovative products, and staffed by the developers and technical experts. By far the most popular new product at the vendor exhibits was "Simon Says" by HSD. Simon Says is a voice recognition macro facility that allows users to assign voice commands to anything from command keys to shell commands. Since every NeXT has a built-in microphone, this is a fantastic addition. There is nothing more satisfying than being able to tell your computer where to go. The voice recognition system is speaker dependent not only for simplicity but also for security. Visus has a fully speaker independent voice recognition system available. Other highlights of the show are posted to the *comp.sys.next* newsgroups.

I didn't have the opportunity or the desire to attend the user or developer conferences at the NeXT Expo. The user conference provided a variety of 30-some hour long sessions over two days geared toward the average user. The sessions ranged from getting the most from particular applications to hints/tips and intro UNIX systems and network administration topics. The developer sessions addressed an equally broad spectrum of topics for the average developer and focused on changes in 3.0.

Barbara Dyker works at the University of Colorado in Boulder in the Computer Science department as the Computing Operations Manager and does UNIX systems and internet-working consulting in what little spare time that leaves.

Report on the BSDI BOF at the Winter '92 Conference

Kevin C. Smallwood
<kcs@houdini.cc.purdue.edu>

Editor's Note: This article describes a BOF I hosted at the Winter '92 USENIX conference. It might appear to be a conflict of interest to run it here. After further consideration, I do not believe this to be so. I hope we can publish summaries of all the interesting BOFs at our conferences and workshops.

Rob Kolstad presented a Birds-Of-a-Feather (BOF) Session on the new BSD/386 product from Berkeley Software Design, Incorporated (BSDI). Kolstad first presented BSDI's goals: to enhance "open systems" perception and availability, to increase availability of a truly standard UNIX (without encumbrance of expensive licenses), to enable standard, usable access to powerful, yet inexpensive, hardware, and to leverage and disseminate working Berkeley (BSD and CSRG) software.

Kolstad is the Program Manager for BSDI, a truly distributed company with employees working all over the country from Falls Church, Virginia, to Berkeley, California. BSDI currently has one manager, five engineers, and one customer service representative. BSDI is clearly a start-up company with all of the employees wearing several hats and putting in long hours.

BSD/386 is a Berkeley compatible operating system for the 386 and 486 PC architectures based on the second Networking Release from the Computer Systems Research Group (CSRG) of the University of California, Berkeley. BSD/386 contains no AT&T licensed code, and, thus, is free of AT&T licensing restrictions.

BSD/386 is ANSI and POSIX compatible. BSD/386 includes the full TCP/IP networking, the X11R5 X Window System, a reimplement of Sun Microsystems Network File System (NFS), text processing tools such as "groff" and TeX, many GNU utilities, along with the many standard UNIX tools. Kolstad reported that SCO V.3.2 binary compatibility will be present in BSD/386 sometime in the Fall of 1992; DOS emulation should be present by December, 1992.

BSD/386 supports the most common 386/486 PC hardware devices currently on the market. Most recently, SCSI support was added. Kolstad also related a story about a person who was able to successfully boot BSD/386 on all of the 386/486-based PC's at a ComputerLand store. Specific hardware questions should be directed to BSDI.

BSD/386 is licensed to a single system for \$995, distributed on a boot floppy and QIC-150 cartridge tape. Kolstad said that the full source is included, but binary-only licenses will be available sometime in the third quarter of this year; the cost is expected to be \$500. Furthermore, for sites with multiple PC stations that only require one copy of the source, "binary right to use" licenses are available for \$200 each. Sixty days of support is included with the purchase and additional support is available. Kolstad emphasized several times during the BOF that site licenses and special needs are very negotiable; interested persons should contact BSDI at 719-593-9445 or by sending electronic mail to bsdi-info@bsdi.com.

The "production" release of BSD/386 is expected in June. Kolstad added that there are currently around 30 "alpha" tapes out in the field. At this point in the BOF, Dick Dunn<rcd@eklektix.com> gave a report on his experience as an "alpha customer." Mr. Dunn started out by saying, "BSD/386 is real, it's alpha, it's real alpha." However, he then quickly added that he "can't panic the system without doing something really nasty like subjecting his system to static electricity." Mr. Dunn's comments were all positive and enthusiastic.

After the BSD/386 presentation, the floor was open to questions. Many of the questions clarified some fine points about BSD/386. Someone asked if a 386-based PC had to have the 80387 floating point chip; the answer was that BSD/386 will run fine without the 387, but that it does use the 387 for floating point arithmetic if it is present. Another questioner asked if bug fixes and enhancements of 386/BSD will be allowed on the net; Kolstad said that the copyright and disclosure of BSD/386 source is still being discussed at BSDI.

When asked about minimum hardware needs for BSD/386, Kolstad answered that BSD/386 will run with a minimum of 100 Meg of disk space without the sources on-line, and that BSD/386 would run in 4 Meg of RAM, but 8 Meg is suggested if running the X Window System (and over 256 Meg of RAM is supported). Kolstad emphasized during one question that BSD/386 runs fine on a 386-based PC though a 486-based PC is faster.

Kolstad was then asked about the BSDI source code versus the BSD source code. He stated that the BSDI source code will be well marked and that the BSD source will continue to be redistributable. Another question asked about support of IBM's Micro-Channel; Kolstad said that no support is planned by BSDI, but that a site is working on it. A question about DOS file systems and

UNIX file system coresidency was raised; the response seemed to be that it was possible, but not easy for the faint of heart in the alpha release; it will be supported in production.

Finally, specific questions about BSD/386 should be addressed to BSDI at the above telephone number or electronic mail address.

Election Results

The results of the elections for the Board of Directors of the USENIX Association for the 1992-94 term are as follows:

President:

Stephen C. Johnson
829 + 110 abstentions

Vice-President:

Michael D. O'Dell
848 + 91 abstentions

Secretary:

Evi Nemeth
842 + 97 abstentions

Treasurer:

Rick Adams
605

Directors:

Eric Allman	661
Barry Shein	499
Lori Grob	491
Tom Christiansen	426

Not Elected:

For Treasurer:

Ed Gould	304
----------	-----

For Director:

Dr. Daniel E. Geer	373
Sonya Neuffer	342
Greg Rose	316
Trent Hein	271
Nawaf Bitar	142

Total number of ballots cast: 939

Annual Report - 1991

Marshall Kirk McKusick, President
<mckusick@usenix.org>

The past year has been an exciting one for USENIX. With a recession gripping the economy and restricting travel budgets (particularly for such "perks" as attending conferences), and the Gulf War erupting two days before our Winter conference, we knew from the start that 1991 would be a challenging year. By year's end, the Association had record attendance at its Fall conferences, had streamlined its operations to produce a nearly balanced budget, and had even managed to expand member services.

In 1992, USENIX is once again operating in the black and should have enough resources to undertake new and exciting projects in the months and years ahead. During the past year, many new ideas were explored or came to fruition. The parallel track of invited talks were so well received that they became a permanent part of the semi-annual conferences. The Large Installation Systems Administration (LISA) Conference, which continues to fill an important and focused member need, reached an all-time high in attendance. Similarly, the MACH symposium expanded its tutorials, papers, and audience. Three tightly focused workshops -- C++, Symposium on Experiences with Distributed and Multiprocessor Systems (SEDMS), and Software Development Environments in UNIX -- provided smaller venues for more academically oriented groups.

Numerous member services were added or expanded. Student scholarships to attend Usenix events hit an all-time high, with 55 awards being made throughout the year. Cooperation with other groups was expanded; a reciprocal agreement was reached with the Sun User Group similar to the existing agreement with EurOpen to offer discounts for publications and conference registration. USENIX continued to have an Institutional Representative to POSIX, a Watchdog report editor, and a representative to ISO WG15.

Best student and overall paper awards were added to reward authors that put in the extra effort at making their papers useful and accessible to the attendees. Publications continue to be an important part of the Association's member services. The association's flagship publication, *Computing Systems*, not only managed a bumper crop of excellent articles, including the special issue summarizing the most important work presented at SEDMS, but also managed to get back on schedule. The sale of annual subscriptions to all USENIX proceedings continues to grow as individuals and libraries discover their high value. Finally, USENIX has appointed a features editor to this newsletter to begin revamping its contents and expanding its coverage beyond the announcements and standards reports.

The board of directors took several steps to promote the continued health of the Association. A policies document was developed to codify many of the previously unwritten rules of the organization that had neither been clearly codified nor consistently applied. A biennial review of the Association's books by an outside CPA was completed in time for presentation to the newly elected board members so that they will have confidence that they are getting a fair assessment of the financial state of the organization for which they are taking fiduciary responsibility. Finally, the board made a conscious effort to locate future events in attractive and inexpensive venues to encourage attendance.

My two year term as president has been both challenging and exciting. The past two years have covered some turbulent times for USENIX, but the organization has emerged from 1991 leaner and stronger and ready to move aggressively and decisively into the future.

USENIX Association Financial Statements

STATEMENT OF REVENUE AND EXPENSES AND CHANGES IN FUND BALANCE For the Years Ended November 30, 1991 & 1990

REVENUE	1991	1990
Membership Dues	\$ 303,165	\$ 236,090
Product	144,914	136,382
Conferences	1,562,244	1,478,821
Exhibition	99,591	159,484
Professional Dev. Seminars	0	34,435
Interest	69,405	92,344
Other	4,082	3,239
Total Revenue	\$ 2,183,401	\$ 2,140,795
EXPENSES		
Membership Services/General Admin.	\$ 674,344	\$ 623,675
Exhibition	86,423	214,183
Conference	1,115,395	1,076,686
Professional Dev. Seminars	0	31,161
Newsletter & Journal	113,744	134,774
Products	69,991	73,535
Projects	102,186	98,141
Depreciation	34,814	40,496
Total Expenses	\$ 2,196,897	\$ 2,292,651
Excess Revenue Over Expenses	\$ (13,496)	\$ (151,856)
Fund Balance Beginning of Year	\$ 1,207,196	\$ 1,318,587
Fund Balance End of Year	\$ 1,193,700	\$ 1,166,731

BALANCE SHEET As of November 30, 1991 & 1990

ASSETS

Current Assets		
Cash	\$ 938,741	\$ 976,007
Accounts Receivable	12,337	0
Prepaid Expenses	94,443	72,767
Inventory	45,362	48,542
Note Receivable	0	115,000
Total Current Assets	\$ 1,090,883	\$ 1,212,316
Fixed Assets		
Securities	\$ 191,601	\$ 0
Net Property and Equipment	51,551	81,096
Total Fixed Assets	243,152	81,096
Total Assets	\$ 1,334,035	\$ 1,293,412

LIABILITIES & FUND BALANCE

Current Liabilities		
Accrued Expenses	\$ 84,026	\$ 65,881
Deferred Revenue	56,310	20,335
Total Liabilities	\$ 140,336	\$ 86,216
FUND BALANCE	\$ 1,193,700	\$ 1,207,196
TOTAL LIABILITIES & FUND BALANCE	\$ 1,334,036	\$ 1,293,412

STATEMENT OF CASH FLOWS For the Years Ended November 30, 1991 & 1990

	1991	1990
Excess Revenue Over Expense	\$ (13,497)	\$ (111,391)
Depreciation	\$ 34,814	\$ 40,496
Increase in Short Term Receivables	(12,337)	0
Decrease/Increase in Inventory	3,180	(24,180)
Increase in Prepaid Expenses	(21,676)	(33,900)
Increase in Accrued Expenses	18,145	28,603
Increase/Decrease in Deferred Revenue	35,975	(10,075)
Total	\$ 58,101	\$ 944
Total Net Operating Cash	\$ 44,604	\$ (110,447)
Increase in Long Term Securities	\$ (191,601)	\$ 50,000
Decrease in Note Receivable	115,000	
Increase of Property and Equipment	(5,269)	(35,599)
Total	\$ (81,870)	\$ 14,401
NET CHANGE IN CASH	\$ (37,266)	\$ (96,046)

Board Meeting Summary

Ellie Young, Executive Director

Below is a summary of the actions taken at the regular quarterly meeting of the USENIX Board of Directors held in Colorado Springs, Colorado on March 31, 1992.

Attendance:

Rick Adams, Ed Gould, Rob Kolstad, Kirk McKusick, Sharon Murrel, Evi Nemeth, Mike O'Dell, Barry Shein, Ellie Young, Judith DesHarnais, Eric Allman, Tom Christiansen, Lori Grob, Steve Johnson

SEDMS III

O'Dell said the symposium was quite successful and has gained a lot of respect in the academic world. There were 100 attendees and a large number of first time presenters. The attendee survey results revealed that: 12 - 18 months is the right frequency, and the academic attendees would like more industrial/commercial participation.

SEDMS IV

Young would work with the program chairs on choosing a date, and that it not be scheduled close to the Mach Symposium in 1993.

Proposals to Chair LISA '93

Kolstad went through the four proposals received and gave a brief summary of each. The proposal made by Bjorn Satdeva was approved.

Proposal for a UNIX Applications Development Workshop

Kolstad explained that the proposal from Greg Woods would have a large number of features and will have more invited presentations when compared to our other workshops. UniForum Canada would be the co-sponsor. It was decided to accept the proposal, with Allman and Kolstad serving as co-liaisons to the program chair.

Budget

Young went over the year-end statements for 1991 (see page 12 in this issue). Young asked for the Board's advice on how to achieve their goal of putting \$150,000 into the Reserve each year.

After a lengthy discussion regarding the budget process and various models for achieving this goal, Young was asked to provide a model/analysis based on increased revenues.

Executive Director's Report

The following issues were decided: that funds be allocated to the budget to upgrade office computing, and the executive committee would oversee this project; that the Executive Director and Conference Coordinator will provide guidance to the various individuals regarding reasonable meal expenses; and that the mailing list rental fee be set at a market competitive rate as assessed by the Executive Director from time to time.

Online Library/Index

Shein reported that he had input most of the programs from all the past conferences and workshops. Young said her staff could handle inputting other publications' contents and future programs.

Book Program

O'Dell and Young discussed the idea of launching a series of the best USENIX papers from the journal and proceedings in specific, focussed areas such as C++. The challenge would be to find people to perform the editorial selection. It was decided to unfreeze the budget line items for the book program, and develop the greatest hits series of USENIX publications.

Login: Report

Kolstad gave a report on the contents of the upcoming issue which would feature articles and reviews that are light and breezy. After an initial couple of issues, the membership would be surveyed to get feedback on these new features.

SIGs

Adams handed out a draft document for the development of USENIX SIGs. It was suggested that a LISA SIG would be a logical first, and the BAY LISA group had already discussed their interest with the Executive Director. Adams explained that the benefits to USENIX of having SIGs would be to increase the membership, and offer more services for those people. It was decided to delegate exploring and negotiating with the BAY LISA group to Johnson, with the target of getting something in place by June. The board should send comments on the draft document to Adams. He and Johnson will then incorporate them along with feedback from LISA group and conclude this by e-mail.

World Forum

O'Dell said the assemblage of world groups at its January meeting was interesting, and the groups wanted to cooperate and communicate in other ways besides political ones. USENIX suggested that they collect names of newsletter editors and create a wire service electronically for sharing information among Unix-related organizations. This service would contain regional reports, events, features, calendar, solicitations, and speakers. We would assist in setting up a mailing list and attempt to address the problem of organizing technical connectivity.

Next Meeting

It was decided that the next meeting will be held in conjunction with the San Antonio conference on June 7. The Annual Meeting of the Association (the Open Meeting of the Board of Directors) will be held on June 11 from 1-2:00 p.m.

How to Contact the USENIX Board of Directors

If a member wishes to communicate directly with the USENIX Board of Directors, he or she may do so by writing to board@usenix.org. A list of the USENIX Board of Directors along with their individual e-mail addresses is contained in page 2 of this newsletter.

Prime Time Freeware

Prime Time Freeware CD

USENIX has arranged for a special discount for its members for the first issue of Prime Time Freeware. This CD contains over 1,500 MB of UNIX-related source code. The disc contains compressed tar files in the ISO-9660 CD-ROM format. A 50 page booklet introduces and explains the disc. Over 100 packages comprise Volume 1 Number 1, including:

Andrew windowing code
Athena (except Kerberos)
CLU
Epoch
GNU (current and vintage versions)
Icon InterViews
ISODE
Kermit (tapes A-E)
Mach NCSA Data Analysis Code
`comp.sources.{3b1,amiga,sun}`
`comp.sources.{games,unix,x}`
Scheme, T
Serpent
Utah Rendering Toolkit
UnixTeX X11R5

Order the disc and booklet directly from:

Prime Time Freeware
415-112 N. Mary Ave., Suite 50
Sunnyvale, CA 94086 USA
+1 408 738-4832
<cfcl!ptf@apple.com>

The discounted price for USENIX members is \$50 (quantity 1-9) plus domestic shipping of \$5/order (\$10/order for foreign shipping). California residents should add 7% sales tax. Orders may be paid, in US funds, by check (payable through a US bank), money order, or credit card (Visa/Mastercard).

Contact Prime Time Freeware for details on larger orders or for other arrangements. Orders may be paid by check, money order, or wire transfer.

Please contact me if you have any questions or problems about the above changes.

Yours,
Rich Morin
<cfcl!rdm@apple.com>
415-873-7841

Report from EurOpen

Alain Williams
Editor of EurOpen Newsletter
<addw@phcomp.co.uk>

The EurOpen/USENIX Conference held April 6-9 in Jersey, Channel Islands, was a successful and interesting event, with portability being a strong theme. The keynote speaker, Mike Banahan, who presented the point of view that most UNIX tekies have an obsession with software portability, whereas most real end user's views are that data portability is far more important. He was followed by speakers who looked at portability from a variety of points of view ranging from "How we do it" to "How to insulate yourself from changes in fashion of output device." Presenters included: Donald Lewine, Brian O'Donovan, Joseph Arceneaux, Andrew Hume, and Barry Shein.

The proceedings are now available from the EurOpen Secretariat at a cost of 25 or 35 pounds/ECUs (members) and 65 or 95 pounds/ECUs (non-members).

U.S. readers may wonder what an ECU is, or be under the impression that it went out with France's Louis XV. An ECU is a European Currency Unit. The basic idea was to have a common currency throughout the European Community, and that this would make inter-European commerce easier. Unfortunately, no member of the European Community was willing to accept the idea that any country other than their own could be the common currency (I personally can't understand why they failed to make the obvious choice of using the British Pound). Thus the ECU was born. It is now several years old, and has recently been available as coinage. Most shops still won't accept it, and most banks insist on treating it as a "foreign" currency which thus provides them with the excuse of levying another charge when you use it. It is worth (roughly) one U.S. dollar.

Future Conferences

A major event in the Autumn that you should not miss is OpenForum '92. It is jointly organized by EurOpen and UniForum, and takes place in

Utrecht, The Netherlands on November 25-27, 1992. This will be a major exhibition and conference of pan-European relevance. It will be *the* European Open Systems show.

Please note that the date for the 1993 EurOpen conference in Seville, Spain has been moved back a week in order to prevent a clash with Feria, a Spanish carnival week. It is now scheduled for May 3-7. I suggest that you book your holidays now so that you can go to the Feria and then the conference!

EurOpen Governing Board

The EurOpen Spring Governing Board meeting was held on April 4 and 5, 1992 in Jersey, Channel Islands. There were 44 attendees representing 22 National Groups and the Executive Committee. Countries represented were: Algeria, Austria, Belgium, Czechoslovakia, Denmark, Finland, France, Germany, Hungary, Iceland, Ireland, Italy, Netherlands, Norway, Portugal, Russia, Spain, Sweden, Switzerland, Tunisia, UK, and Yugoslavia.

An application for membership from the Luxembourg UNIX User Group was accepted. Ongoing actions discussed at the meeting included proposals for different types of future events and educational workshops; investigations of whether readers of the EurOpen newsletter might be interested in receiving abstracts in English of interesting papers from other National Groups; examining the possibilities of holding directories on central site; creating a framework which National Groups can establish with the Backbones; and examine the possibilities of a Belgian Law legal umbrella for EurOpen.

During the meeting voting took place for the Chairman and new Executive Committee using the single transferable vote and the following were elected:

Michel Gien, Chairman
Frances Brazier
Norman Hull
Ernst Janich
Nigel Martin

Thanks were expressed to Kim Biel-Nielsen whose nomination for election was unsuccessful, but who has done so much work as a member of the committee in the past. The Governing Board expressed disappointment that no new nominations had come forward from the National Groups. To encourage new blood, the Executive Committee agreed that its mandate would be for one year only and that new elections will take

place at the Spring 1993 meeting.

The overall organization, management, and role of the Executive Committee is undergoing changes since much of the work previously done by them is passing to sub-committees of the Governing Board and the Executive Director, Pierre Scheuer.

Four sub-committees have been set up to: examine income strategy and fees; help with better forward planning and accountability of the different activities of EurOpen; look at the implications of moving under Belgian Law; formulate a future perspective for the Newsletter; formulate the needs and requirements of EurOpen Users for networking on the medium and long term; and propose the directions for future EurOpen events.

The sub-committee members are drawn from those with relevant experience and interest in the national groups. The use of such members helps to reduce the large work load of the EurOpen

executive committee, and will bring in fresh new ideas. It is hoped that the fruits of this initiative will be a EurOpen that serves its members better.

Due to pressure of work Mr. Jean-Michel Cornu has had to stand down as co-ordinator of the EurOpen Working Group, and a new coordinator is being sought. The formation of Working Groups is being actively encouraged within the National Groups, as is the participation of those local Working Groups at the EurOpen level.

Representatives of USENIX were also present at the Governing Board meeting, and Barry Shein expressed the view that it was very worthwhile and important for UniForum, USENIX, and EurOpen to meet together regularly.

The next EurOpen Governing Board meeting will take place during the OpenForum Conference in Utrecht in November, 1992.

Eclectica

"Reflections"

Robert D. Carlitz <RDC@vms.cis.pitt.edu>

Editor's Note: I found this fascinating. Can you imagine watching 24 hours of 93 channels of television?

There's a very interesting article in the March 9 issue of the *New Yorker* by Bill McKibben entitled "Reflections." McKibben writes about television networks, and he raises questions which are relevant for the development of computer networks – especially networks for the education community.

McKibben contrasts a day of television with a day in the wilderness. His TV day is spent on the Fairfax County, Virginia cable system. After taping one day's programming from the system's 93 channels he spent a number of months watching it all and studying it. His wilderness outing was in the eastern mountains.

Several of McKibben's observations are startling. He suggests that television has indeed produced McLuhan's "global village," not by enriching our understanding of the world but by reducing the entire world to a level of common superficiality.

He repeatedly talks about how this technology is destroying information rather than enhancing it.

We may be wise to consider McKibben's arguments as we try to "tame" the anarchy of the Internet for use in school classrooms. It may be that what is best about the Internet is precisely its unpredictability, and that insofar as we try to package the wealth of information that we find amidst this anarchy, we may actually be destroying precisely what we are trying to preserve for others to use.

McKibben makes the point that a major failure of television stems from its need to squeeze everything into a common format. Not only does this cheapen the images distributed via television, but it makes it impossible for viewers to distinguish the importance or quality of what is offered – even when the broadcasters themselves might feel that obvious distinctions do exist.

Advocates of computer networks in education need to answer the question of critics: "How does this technology differ from other technologies which have been introduced to reform education but which have never met their original prom-

ise?" I can think of two answers to this question:

(1) Computer networks allow for active learning, where students seek information via the network and organize it themselves, perhaps with the aid of a range of computer tools.

(2) Computer networks are very flexible. It's important to retain this flexibility in the design of information services on the network.

This attitude fits in well with current design philosophies. The network itself provides a transport mechanism which is independent of the nature of

the data that it carries. Servers provide resources independent of details of the client programs which use these services. User interfaces are customized to fit the needs of individual users and are not limited in any way by either the network or the structure of information servers on the network.

It certainly makes sense for us to draw lessons from other technologies that have been relatively unsuccessful in the school context. McKibben's article provides some thoughtful insights to ponder.

Can UNIX Designers Learn Anything from PCs?

Jeff Haemer
<jsh@canary.com>

Editor's Note: An Open Letter to Marc Rochkind

Dear Marc,

Seeing your Invited Talk abstract in the 1992 Winter USENIX brochure (Can UNIX designers learn anything from PCs?), makes me wonder if I should propose an analogous talk for Comdex: Can PC designers learn anything from UNIX? Here are the first ten reasons I came up with.

- UNIX does more than DOS.

There's a lovely old Ken Olsen quote about how UNIX is simple but VMS is complete. Today, UNIX is also complete. Steve Zucker points out: "In 1980, a single person could read and understand the entire UNIX kernel. In 1990, a single person couldn't lift it." Remember, if simplicity worked, the world would be overrun with insects.

- UNIX makes writing large programs convenient.

Contrast the size of LOTUS 1-2-3 with the size of xclock. Think of how much more Shakespearian literature we'd have if he hadn't been hobbled by the sonnet form and the attention span of the audience at the Globe.

- UNIX permits complex system administration.

A properly written labyrinth of `/etc/rc[0-9].d` directories eliminates most by-hand operations during reboots, so that you can go make lunch while your machine is powering back up and putting all your files in lost+found. Appliances belong in the kitchen, and sometimes so do you.

- UNIX provides good programmer support.

200 PCs only create a job for a single system administrator. 200 Suns create jobs for at least eleven: ten administrators *and* a system administration manager.

Contrast the average salary of device-driver writers in the UNIX and DOS worlds, and you'll see why UNIX vendors brag about the dozens of devices and displays they support, while your latest DAK catalogue begs you to take their surplus: "\$1,548.95 worth of CD-ROM software for just \$149." Sure, you have to buy a \$200 CD ROM drive, too, but you can bet it comes up as soon as you plug it in.

- UNIX encourages portability across platforms.

DOS programmers were stuck with function prototypes faster than X3J11 could decide what it thought it might maybe think about doing. Because of the close relationship between C and UNIX, UNIX takes a more conservative approach.

Converting programs to ANSI is error-prone and you often have to debug the result, which means either putting in lots of printf statements or getting out the sdb manual. My vendor-supplied UNIX "ANSI C" compiler still generates big, slow code, and sets *argc* to 0, however many arguments you pass it. Things like this encourage use of the standard, UNIX "Portable C Compiler,"

/bin/cc, which allows you to continue to compile your code on PDP-11/70s.

As a result, UNIX provides pretty good source-code portability across such different INTEL UNIX binary formats as a.out, x.out, coff, gpoff, elf, whatever Sun did for 386s, BCS, BCS-2, and the new System-V-consider-them-a-standard-dot-4 ABI. Indeed, source code portability often transcends UNIX. X Windows and OSF/Motif, for example, will soon run on everything from VMS to the Macintosh.

- UNIX is a potentially more lucrative software market than DOS.

Prices of Microsoft Word, Lotus 1-2-3, and the Norton Utilities for UNIX, show that all can have much higher profit margins on UNIX. (Admittedly, some of that difference is inflation, since the proper price comparison is with older DOS versions of comparable functionality.)

- UNIX provides device-independent I/O.

This lets state-of-the-art UNIX word processors, like *ex* and *nroff*, run on a wide range of devices, from teletypes, to terminals connected to 300-baud dial-up lines, to */dev/null*. In contrast, many standard Mac applications, like HyperCard, require mice, and don't work at all on ANSI-standard terminals or VT100s.

The optimizing, screen drawing package, *curses*, lets screen-oriented, state-of-the-art UNIX word processors, like *vi*, run quickly on the same range of devices.

- UNIX comes bundled with a lot of useful applications software.

At my local Office Club store, Reader Rabbit for DOS costs \$20, but the far-more-flexible quiz, from AT&T, is a standard part of many UNIX distributions. Moreover, quiz doesn't restrict me to machines with speakers for audio output.

- UNIX is serious about formal standards.

PCs tend to be drawn into immediate practical solutions and de facto standards. By the time formal ANSI standards are finally established, PC users often have a huge installed base of software that has to be upgraded if it wants to follow the rules. This is why the U.S. Government likes UNIX.

UNIX is also serious about international standards, and standardizing internationalization, and funny character sets, which, as John Quarterman points out, will allow the ls command to have a lot more options. This makes UNIX popular with foreign governments, like Belgium, Japan, X/Open, and what's left of Croatia.

- UNIX is a trademark.

Folks are always using "DOS" and "PC" as nouns, threatening the trademark-protected revenue streams of Microsoft and IBM, and discouraging further investment in software development. In contrast, AT&T has lawyers to help us remember not to omit the "System" from "UNIX System" to "UNIX System CoPy."

This week, on L.A. Law:

See, Benny, this entire letter uses UNIX as a noun instead of an adjective, so it's ungrammatical and, therefore, illegal.

Gee Arnie, maybe that's why my mom always made me use a syntax-directed editor.

Regards,

Jeff Haemer

An Update of UNIX-Related Standards Activities

Stephen Walli <stephe@mks.com>

Report Editor

USENIX Standards Watchdog Committee

Report on the IEEE Standards Board

An Anonymous Friend of USENIX reports on the December 3-5, 1991 meeting in New York, NY :

[Ed. – The report writer asked to remain anonymous. Anyone wishing to send comments to the writer may do so through me. – SW]

The IEEE Standards Board is the committee responsible for overseeing all standards related activities within the IEEE. The IEEE produces standards for the entire electrical engineering spectrum, not just the Computer Society. The Technical Committee on Operating Systems — Standards Subcommittee (TCOS-SS), is the IEEE Computer Society committee responsible for the POSIX family of standards.

As usual, the December 1991 meetings of the IEEE Standards Board produced a plethora of new Project Approval Requests (PARs) and approved projects, some new rules to apply to the standards process, and one more new Organizational Representative that can ballot POSIX standards.

Acknowledgments

Perhaps the new rule that most impacts the IEEE community is one concerning the use of acknowledgment sections in standards. You've probably seen one of these sections before; they're the ones that thank your company/university/organization/mother for providing the means for you to contribute your thoughts and ideas to that lovely thing known as the standards process. It's usually found in the front or back of the standard in what we jargon-savvy folks know are informative sections of the document, so it's not part of the official standard. (Don't confuse it with the foreword or introduction, which discuss the technical and historical development of the standard and list the working group and balloting group.)

The IEEE Standards Board Procedures Committee (whew! that's ProCom for short) felt that the IEEE could be legally liable if the standards mentioned a company without first asking their permission. A policy was proposed that said a working group could include one of these sections if each member obtained written permission from the compa-

nies/etc. involved, to be kept on file with the IEEE Standards Department. There are form letters for you to follow, but nonetheless it's an extra step you'll have to take.

Of course, the question comes up as to whether you should be doing this work at all. What if one company says yes and 20 say no? Do you have an acknowledgments section that only lists a few companies? For a family of standards like POSIX, should some standards have this section and some not? As always, things rapidly get complicated. Because of this, the POSIX technical editors had a lengthy discussion on whether to have these sections at all in their documents. Opinion was wide-ranging and varied; the interim decision was to go to our individual working groups and ask for their opinions. Based on those discussions, the technical editors will decide whether to keep these sections in the future.

The Curse of Acronyms

As we all know, standards-writing groups have a seemingly inexhaustible ability to create acronyms. Indeed, after a while our conversations seem to consist entirely of abbreviations, and woe betide the person who tries to understand our arcane internal code.

Of course, the Standards Board has to do just that when they look at our PARs (oops! that's Project Authorization Requests). They understandably get frustrated. Because of that, the New Standards Committee (NesCom) has said that they don't want to see incomprehensible acronyms on PAR submissions in the future. The NesCom members come from all societies of the IEEE, not just Computer, and many power-engineering standards developers can't begin to guess at what an acronym means that you've used since the first time you touched a keyboard.

This means we'll have to get used to standards titles that are even longer than they are now! When filling out a PAR, you'll have to remember to expand acronyms appropriately. You wouldn't want to have the PAR rejected on these grounds. This subject will be discussed in more detail at the next NesCom meeting.

One Man, One Vote

Questions have arisen as to whether or not members such as Institutional Representatives and similar reps in the power engineering realm vote twice on a document, once as an individual and possibly again representing their organization. The Board agreed to appoint an ad hoc committee to look at the issue of one man, one vote. More information should be available from forthcoming meetings.

In other IR related news, SPARC is now officially approved by the IEEE Standards Board as an IR and has the right to vote on all POSIX documents.

[Ed. — The following lists are provided to allow the reader to appreciate the full breadth of control the IEEE Standards Board has as its mandate. These are still just the Computer Society related standards. The reader should note P1279, P1281, and P1282. Andrew Hume regularly warns in his ANSI WORM standards reports that the WORM standards may have a far broader impact than people think. Here, in P1282, we even see them "worming" their way into POSIX.1 (ISO 9945-1).]

And here's the information on Review Committee (RevCom) and NesCom Computer Society activity:

Approved New Computer Society PARs

P1278 (C/SCC) Standard for Information Technology-Distributed Simulation Applications-Process and Data Entity Interchange Formats

P1279 (C/SCC) Standard for Information Technology-CD-ROM Architectural Profile

P1281 (C/SCC) Standard for Information Technology-Use of ISO 9660: 1988 System Use Fields

P1282 (C/SCC) Standard for Information Technology-Interchange of ISO 9945-1:1990 Filesystems via the ISO 9660: 1988 File Structure

P802.1j (C/TCCC) Standard for Managed Objects for MAC Bridges (Supplement of 802.1D)

P802.1k (C/TCCC) LAN/MAN Management Information for Monitoring and Event Reporting

P802.2C (C/TCCC) PICS Proforma for LLC Type 1 Operation and LLC Type 2 Operation

P802.1D (C/TCCC) Technical and Editorial Corrections to Std. 802.1D

P802.2f (C/TCCC) Standard for LLC Sublayer Management

P802.6k (C/CC) Distributed Queue Dual Bus Subnet work of a Metropolitan Area Network, Supplement for MAC Bridging

Revised Approved Computer Society PARs

P1209 (C/SE) Recommended Practice for the Evaluation and Selection of CASE Tools

P802.1F (C/CC) Common Definitions and Procedures for 802 Management Information

P1155 (C/MM) Standard for VMEbus Extensions for Instrumentation: VXIbus

P1175 (C/SCC) Trial Use Standard Reference Model for Computing System Tool Interconnections

P1396 (C/MM) Standard for a Communication Bus (TELECOM Bus): Reference Models

Withdrawn Computer Society PARs

P1101.5 (C/MM) Standard for Mechanical Core Specification for Microcomputers-Desktop Form Factor

Approved New Computer Society Standards

610.6 (C/SCC) Standard Glossary of Computer Graphics Terminology

1029.1 (SCC20 & C/DA) Standard for Waveform and Vector Exchange (WAVES)

1175 (C/SCC) Trial Use Standard Reference Model for Computing System Tool Interconnections

P1212 (C/MM) Standard Control and Status Register (CSR) Architecture for Microcomputer Buses

Withdrawn Computer Society Standards

IEEE Std 662-1980, IEEE Standards Terminology for Semiconductor Memory (ANSI)

Report on POSIX.0: The Guide to Open Systems

Kevin Lewis <klewis@gucci.enet.dec.com> reports on the January 13-17, 1992 meeting in Irvine, CA:

The POSIX.0 working group adjourned the October meeting wondering what the mock ballot would yield. This uncertainty was focused not only on the size of the return, but also on whether there were any hidden or significant issues lurking in the darkness.

Twenty six mock ballot responses were received: 13 users, 9 producers, and 4 general interest participants. This reflects a healthy balance. In total, there were approximately 760 objections/comments. Some ballots covered specific sections, while others addressed the entire guide.

It appears that the issue of "public specifications" that has been lurking around in other venues has arisen here. For those of you not familiar with this, I cannot do it justice here. Suffice it to say that it involves the use within public procurements of specifications that are not currently in the formal standards process but which have widespread industry use and acceptance.

POSIX.0 feels that such specifications are acceptable under specific conditions which include consensus, availability, lack of encumbrances, and proper documentation. (There is much, much more to this, so get a copy of the guide or call someone in POSIX.0 if you are interested or concerned.)

The decision was made in January to move forward on the formal ballot. POSIX.0 has notified the IEEE and a letter forming the formal ballot group will go out in the March-April time frame. The goal is to begin the formal ballot in July. In parallel, POSIX.0 will be submitting the guide to the international standards community in order to obtain review and comment and to prepare the way for it as an ISO Technical Report.

Report on POSIX.2: Shell and Utilities

David Rowley <david@mks.com> reports on the January 13-17 meeting in Irvine, CA :

Summary

The end is in sight. POSIX.2 (Shell and Utilities) Draft 11.2 closed its recirculation ballot last October 21. Draft 11.3 is due out any day now. A full draft (Draft 12) will be recirculated to the IEEE working group before the final standard is adopted. POSIX.2a (UPE) Draft 8 closed its recirculation ballot on January 24. Both standards are expected to be approved as full-use IEEE standards at the September meeting of the IEEE Standards Board.

Work on POSIX.2b continues, including the contentious new file format for PAX and extensions to the POSIX.2 utilities to handle symbolic links.

The first cut at test assertions for POSIX.2 has been wrapped up, and assertions for POSIX.2a have begun.

Background

A brief POSIX.2 project description:

POSIX.2 is the base standard dealing with the basic shell programming language and a set of utilities required for the portability of shell scripts. It excludes most features that might be considered interactive. POSIX.2 also standardizes command-line and function interfaces related to certain POSIX.2 utilities (e.g., `popen()`, regular expressions, etc.). This part of POSIX.2, which was developed first, is sometimes known as "Dot 2 Classic."

POSIX.2a, the User Portability Extension or UPE, is a supplement to the base standard. It standardizes commands, such as `vi`, that might not appear in shell scripts, but are important enough that users must learn them on any real system. It is essentially an interactive standard, and will eventually be an optional chapter to a future draft of the base document. This approach allows the adoption of the UPE to trail Dot 2 Classic without delaying it.

Some utilities have both interactive and non-interactive features. In such cases, the UPE defines extensions from the base POSIX.2 utility. Features used both interactively and in scripts tend to be defined in the base standard.

POSIX.2b is a newly approved project which will cover extensions and new requests from other groups, such as a new file format for PAX and extensions for symbolic links.

Together, Dot 2 Classic and the UPE will make up the International Standards Organization's ISO 9945-2 – the second volume of the proposed ISO three-volume POSIX standard.

POSIX.2 Status

Hal Jespersen, Chair of POSIX.2, is about to send out Draft 11.3. This is likely the last "changes-only" draft of POSIX.2.

The POSIX.2/D11.2 recirculation ballot closed October 21, and resolution of ballot objections has completed.

Balloting of Draft 11.2 has been held open pending the arrival of ISO comments. All changes for the next draft (11.3) will be forwarded to ISO through the US TAG.

It is expected that a final draft 12 of POSIX.2 will be made ready in time for the May WG15 meeting in New Zealand, and should be approved as a Draft International Standard.

The technical content of the standard has more or less stabilized. Most recent changes relate to clarifications in wording.

POSIX.2a Status

POSIX.2a is also coming down the home stretch, as the technical content has stabilized. Ballot resolution for POSIX.2a (UPE) Draft 8 was completed. The ballot closed on January 24. The next draft will likely be a quick "changes-only" recirculation, labelled draft 8.1. It should appear any day now.

The ISO ballot ends in April. All comments will be rolled into a Draft 9 which will be produced in time to be carried to ISO in May for approval as a Draft International Standard (DIS).

Hal Jespersen expects that both standards should be given final full-use IEEE approval at the September meeting of the IEEE Standards Board.

Internationalization Inadequacies

Randall Howard, President of MKS, put forward a proposal to the POSIX.2b working group to define a system API to the internationalization information embodied in a POSIX.2 locale. Routines to access collation elements, detect membership within a character class and extensions to the `strftime()` call were presented. The group felt that since it was a system API, not a utility, it rightfully belongs in POSIX.1. When the same presentation was given to POSIX.1, they expressed the opinion that parts of the proposal were better suited to the ANSI or ISO C Standard efforts. Unfortunately, they don't necessarily want it since they haven't (yet) adopted the POSIX.2 definition of a locale. This all demonstrates that the POSIX process cannot effectively deal with issues that cut across a number of working groups and/or standards. Perhaps the Systems Interface Coordination Committee (SICC) that has recently been formed within POSIX can help address some of these issues.

Comments on ISO 10646

The ISO working group that is responsible for the ISO 10646 character set standard (which now includes the Unicode work,) has asked the POSIX.2 working group for their opinion on their current proposal.

The working group expressed much concern over the use of null octets within the valid character codes. Since computer languages such as "C" make use of nulls as a string termination marker, a lot of existing code would have to be heavily modified in order to support the new standard. The working group was against the proposal for this reason. Apparently the ISO/ANSI C working group has expressed similar concerns.

Symbolic Links

Dawn Burnett from USL submitted a proposal on extending the POSIX.2 and POSIX.2a utilities to support symbolic links, based on the System V implementation. The problems that arise from symbolic linked directories were discussed at length. There is nothing more irritating than changing to a directory, printing the current working directory only to find that you have been magically warped to a completely different spot in the file system. A proposal to maintain both physical ("Where am I") and virtual ("How did I get here") paths was offered. The text will find its way into the next draft of POSIX.2b for further discussion.

Test Methods

Real progress was made completing the remaining test assertions for POSIX.2, and beginning the POSIX.2a work. A style guide for writing consistent assertions has yet to appear, but the group seems to have found its stride and is working well.

Test assertions for the interactive utilities have yet to be tackled, but it is expected that it will not be as difficult as first anticipated. The assertions for `vi`, `talk`, etc. will describe (in precise English) what action must take place upon the stated input. The process whereby the results are verified will be left up to the test suite implementor.

New PAX Archive Format

Work continues on the new PAX archive format. A consensus is (slowly) starting to brew. The issue of supported filename code sets is a thorny one, especially since POSIX has not addressed any code set issues in a general way (such as adopting the X/Open `iconv` utility and API).

The problems stem from wanting to use the format to address both universal archive transportability as well as local file system backup and restore, one concentrating on a standard common ground, the other wanting the flexibility of representing the full local filename character set. This is the most contentious area of the format, and there will likely be much wailing and gnashing of teeth before the dust settles.

If you have any interest in this area, the group would be pleased to hear from you.

Report on POSIX.3: Test Methods and Conformance

Andrew Twigger <att@root.co.uk> reports on the January 13-17, 1992 meeting in Irvine, CA:

SCCT Matters

The Steering Committee on Conformance Testing (SCCT) met three times during the week and discussed a broad range of testing related issues. The major issues centered around fitting the test methods into the document structure, dealing with options in "base" standards, and test methods for profiles.

The higher level of document structure seems to have been resolved by introducing a parallel set of documents (and therefore project requests, or PARs) to the base standards. The test methods

documents will be numbered by adding 1000 to the base standard number, i.e., POSIX.6 Security Extensions (P1003.6) will have test methods in a document numbered P2003.6. The IEEE would then resolve any accompanying publication issues.

The more granular issue of how to write assertions which can be easily merged along with the base standard was also briefly discussed, but not yet concluded. The integration of base standards (POSIX.1, POSIX.4, POSIX.6, etc.) is one of the major problems facing TCOS at the moment, but the solution seems as far away as ever. (The Technical Committee on Operating Systems – Standards Subcommittee, TCOS-SS, is the IEEE Computer Society TC responsible for the POSIX standards.)

From the test methods perspective, integrating assertions for a pervasive interface like `open()` introduces a considerable problem in defining which assertions relate to which base standard options. While solutions can be produced easily, these are generally inelegant.

The options issue, which was left over from the Parsippany meeting was readdressed with some further input from POSIX.1. The problem may not be as serious as previously thought and many of the issues can be resolved with some minor changes to POSIX.3.1 (POSIX.1 Test Methods). The remaining ones can be resolved by introducing an announcement mechanism, which most test suites have to provide, allowing the test suite to determine the implementation's option setting.

The SCCT reviewed the meaning of profile conformance and the use of conformance statements in profiles. They agreed that profile conformance statements should refer back to those in the base standards and should be validated by reference to the test methods for the base standards, where available, plus the specific test methods for the "mortar" defined in the profile. (The Profiles Steering Committee is reaching agreement on the rules for subsetting base standards, and how additional behaviour may be thought of as the "mortar" binding the standards together.)

Software Testing Environment BOF

On Monday evening BSI (the British Standards Institution) and Mindcraft called a Birds-of-a-Feather gathering to explain Software Testing International and the Software Testing Environment (STE). Software Testing International would be a non-profit organisation set up to administer the development of test suites for POSIX and other standards. Most of the attendees seemed reticent in their approval of the scheme, particu-

larly when it became evident that Mindcraft would be the sole test suite authoring organization with a seat on the Board. Comments from the presenters that "POSIX testing is just starting to become serious" were also not well received. It seemed clear that both structural and perceptual changes would be necessary before the proposed scheme could make an impact in the POSIX testing arena.

The actual STE introduces an additional API layer on top of the current Test Environment Toolkit (TET), a freely available testing harness created jointly by X/Open, Unix International, and the Open Software Foundation. Initial impressions were that the main purpose of this layer is to allow Mindcraft's CTS based test suites to execute in the TET environment. (NIST is currently supporting the TET as their testing methodology of choice.)

Mindcraft promised to make the specifications available shortly and to provide an implementation at the end of quarter two. The testing community review the value of these extensions, but with significant aspects like distributed testing omitted it may not capture many peoples' imagination.

POSIX.3

The POSIX.3 working group continued in their relentless task of writing and reviewing assertions for the POSIX.2 (Shell and Utilities) standard. The latest draft (POSIX.3.2/D7) has been circulated for review and comment, though no comments have yet been received. At the end of the Irvine meeting it was expected that there would be no significant parts of POSIX.2 that were unaddressed by test methods, except its internationalisation aspects. The working group commenced the specification of test methods for POSIX.2a (UPE) towards the end of the meeting.

Other working groups were also developing test methods for their standards with progress being made by (at least) POSIX.6, POSIX.8, POSIX.12, 1224 and POSIX.17, as well as some of the profile groups. In general, these groups were developing a reasonable understanding of the task facing them, and in some cases good quality test methods have already been produced.

The question of language independent test methods was discussed in the POSIX.1 forum, though other groups (for example 1224) have also made progress in this area. The outcome of the POSIX.1 discussion was an estimate by a prospective contractor to undertake 2,000 or more hours of work to produce LIS test methods for POSIX.1. This looks like an exceedingly high estimate, and I would be very surprised if TCOS followed it up!

Report on POSIX.6: Security Extensions

Charisse Castagnoli <charisse@sware.com> reports on the January 13-17, 1992 meeting in Irvine, CA:

This was the first meeting of the POSIX.6 group since the ballot closed on January 6, 1992. Of the 232 official ballot members, 181 members responded. The response equals 78% of the ballot pool. (A minimum of 75% response is required by IEEE for a ballot to be considered valid.)

The 181 returned ballots were divided as follows:

Affirmative	Negative	Abstain
69	61	51
53%	47%	<don't count>

In order for a ballot to pass, there must be a 75% affirmative ballot. One would think this means 75% of the responses must be affirmative, but this is not the case. Only 75% of the non-abstaining votes need to be affirmative. Taken to an extreme, this means that regardless of the ballot pool size, if three people vote affirmative, 1 votes negative and the rest abstain, the initiative passes. The moral of the story is: abstain only as a last resort, there may be deleterious side effects.

The POSIX.6 committee is now divided into 3 groups: test assertions, new projects, ballot technical reviewers.

The test assertions group, led by David Rogers, is developing the companion document of test assertions. This is required to actually complete the ballot process.

The new projects group is working on new Project Authorization Requests (PARs). Three PARs were presented: one PAR for Identification and Authentication, one for data interchange, and one for terminal I/O.

In addition, PARs were prepared for the existing POSIX.6 functions. The current PAR for the existing functionality will eventually be transformed into POSIX.6a (Security Extensions to System Interfaces) and POSIX.6b (Security Extensions to System Utilities and Shell). [ed. — PARs are essentially administrative project control documents, but are becoming administrative nightmares in the IEEE standards development process.]

The ballot resolution group began reviewing the ballot objections. A preliminary analysis indicated that one common objection was lack of consistency within the ballot. Requests for consistency in function naming, calling parameters, data types, and return codes were frequent. After careful reflection, the ballot resolution group

agreed this was a reasonable request and began to work out a set of guidelines to ensure consistency throughout the draft.

Highlights of the ballot resolution group discussions include:

Should "set" and "get" be used for function names instead of "read" and "write?"

Should data types be contiguous in memory? (That is, can a data object be copied with a bcopy()?)

Should functions manage data storage and allocation or should the programmer manage them?

After arduous negotiations, the group developed a set of guidelines that resolved many issues that have plagued the drafts for years. The ballot resolution group will now join the State Department to support peace negotiations in the Middle East.

The ballot resolution group tested the guidelines by applying them to each of the primary functions in the draft. These functional areas are privilege mechanism, mandatory access control (including information labeling), and access control lists. The auditing functions were granted an exemption from this exercise, because they were being reviewed in light of the new data type guidelines and substantial interface modifications were expected.

Chris Hughes presented some options for new auditing interfaces. The existing interfaces, in addition to being inconsistent, lack good support for application level auditing. Additional work is needed on the auditing functions, and will be presented at the next POSIX meeting.

At the end of the meeting, we all agreed to try and complete the interface changes necessary to bring each function in line with the new guidelines. We also agreed to resolve as many ballot objections as possible before the April meeting.

Report on POSIX.14: Multiprocessor Profile

Rick Greer <rick@ivy.isc.com> reports on the January 13 - 17, 1992 meeting in Irvine, CA:

The multiprocessor working group plans to submit their draft profile to a mock ballot after the April 1992 meeting. Much of the January meeting was spent dealing with various trivialities in the draft in anticipation of the mock ballot. We did, however, encounter one major issue that could prevent the draft profile from ever becoming a standard. It seems that a draft profile cannot become a "POSIX Standard Profile" if it references

documents which are not themselves official standards endorsed by a "recognized accrediting body." The POSIX.14 draft references both the POSIX.4 (Real-time) and POSIX.4a (Threads) drafts, as well as some ongoing ANSI X3H5 work defining parallel language facilities. It cannot become a standard profile until all of these make their way through the appropriate standardization mill.

The POSIX.14 profile is fairly simple, and likely to be ready for balloting long before its antecedent documents. This forces the POSIX.14 working group into one of a number of possible holding actions:

1. Hold up balloting the POSIX.14 profile until all of the referenced documents become standards. This will leave the working group with very little to do, except perhaps to work with the other groups to try and speed up acceptance of their work. Since most of the POSIX.14 working group are refugees of the POSIX.4a threads wars, there is very little enthusiasm within POSIX.14 for this approach.
2. Go ahead and ballot the POSIX.14 profile, but don't submit it to the IEEE Standards Board for approval until the referenced documents become standards. This gives the working group something to do over the next few months (i.e., work on ballot resolutions). In the long run it will only delay the inevitable: We are likely to run out of ballot objections long before the other documents become standards.
3. There are a number of "missing interfaces" that POSIX.14 would like to see added to POSIX.1 and POSIX.2 but, being a profile group, is unable to specify. What we can do is to recommend to other groups that they incorporate these interfaces into subsequent versions of their documents to better support multiprocessor operation. The general feeling within POSIX.14 is that if we do a thorough job of presenting well defined, well rationalized, multi-processor interfaces, the other working groups should pick them up with little argument (ha!). While waiting for the draft documents referenced by the POSIX.14 profile to become standards, the POSIX.14 working group could devote some effort to defining these missing interfaces.

We pretty much decided to go with holding action number 3 (primarily because it's more fun than items 1 or 2), but this course of action presents problems of its own. If we wish to include

the missing interfaces into the profile, we will have to wait for them to become officially adopted into POSIX.1 and POSIX.2. This would, of course, put us right back where we started: waiting for referenced documents to become standards before the profile itself can be finished.

One way out of this dilemma is to include the missing interfaces in an appendix to the profile itself. Once the interfaces have become recognized standards, we can include them in the normative text in a later revision of the profile.

Report on POSIX.17 - Directory Services API

Mark Hazzard <markh@rsvl.unisys.com> reports on the January 13-17, 1992 meeting in Irvine, CA:

Summary

Once again, the POSIX.17 group made solid progress between meetings, completing all major homework assignments. The week in Irvine was a busy one. The Project Management Committee (PMC) audited POSIX.17 and gave the group a clean bill of health. We also met with POSIX.12 furthering our discussion on a simplified API to the directory. Our Mock ballot input on the networking section of the Guide, POSIX.0 Draft 14, were reviewed with POSIX.0, with the promise that they will be reflected in the next draft. We completed processing input from our Mock Ballot of POSIX.17 Draft 2.0 and began drafting responses to our reviewers. We also identified work items and continued planning for an official IEEE ballot which begins April 7, 1992.

Introduction

The POSIX.17 group is generating a user to directory API (e.g. an API to an X.500 Directory User Agent). We are using the joint XAPIA- X/Open Directory Services specification (XDS) as a basis for work. XDS is an object oriented interface and requires a companion document, X/Open's Object Management specification (XOM) for object management.

XOM is a stand-alone specification with general applicability beyond the API to directory services. It will also be used by IEEE P1224.1 (X.400 API), and possibly other POSIX groups, and is being standardized by P1224. Draft 4 of P1224 has already entered IEEE ballot.

POSIX.17 is one of five "networking" groups that currently make up POSIX Distributed Services and as such, POSIX.17 comes under the purview of the Distributed Services Steering Committee (DSSC).

Status

The group chair was unable to attend the meeting in Irvine, CA, so yours truly once again assumed the duties of chair. There has been a low grumble about the ever increasing overhead associated with TCOS/POSIX working groups, and now I know why. A Project Management Committee audit Sunday morning, 2 Sponsor Executive Committee (SEC) meetings, 2 Systems Interface Coordination Committee (SICC) meetings, 2 Distributed Systems Steering Committee (DSSC) meetings, 2 Distributed Systems (DS) Plenaries, a Logistics meeting, a Distributed Security study and (I almost forgot) POSIX.17 working group meetings made for a noticeable lack of "spare" time.

Commitment within the group remains strong, with all other core members attending, and completing their "homework" assignments.

The TCOS Project Management Committee held the first audit of POSIX.17 on Sunday morning. The PMC recommended continued sponsorship of the work, splitting the work into two projects, increasing the size of the working group for ballot resolution and bringing our Issues Log current.

During the week, the group completed processing the comments received from our Mock ballot. We began to draft written responses which will be sent to all who took time to review the draft and provide us with comments and/or objections. Several of the comments/objections resulted in improvements to the specification and will be incorporated into the next draft (3.0). This will be the draft that goes directly to IEEE ballot on April 7th.

The Technical Editor completed the Language Independent Specification (LIS) and a first cut at test assertions as well. (X/Open followed through with their promise to fund our technical editor to write the assertions for POSIX.17. This made sense in that X/Open needs to have assertions for XDS.) The test assertions were reviewed by a consultant from POSIX.3 who had some problems with the way things were done. A lively debate ensued, but in the end, we caved in, and will incorporate the "suggestions." It is estimated that 90% of our assertions will require change. Hopefully, this can be automated.

Once again, we met with POSIX.12 (Protocol Independent Interfaces) in joint session and discussed their requirements on directory services. The POSIX.12 group wants a simplified interface to directory services for the users of their Detailed Network Interface (read sockets/XTI). We also

discussed what objects POSIX.12 will need to be stored by the directory and how those objects will get documented. Given our need to freeze our draft for ballot and the lack of definition for both new objects and interface functions, we explored possible avenues for proceeding with the work.

We met in small group to continue the discussion. POSIX.17 participants left the meeting with a greater understanding of the issues, but no closer to a solution. We had a debriefing session afterwards and decided to produce a white paper documenting agreements, assumptions, issues, options, and proposed actions. This will be used to focus discussion at the next small groups meeting in April.

POSIX.17 and P1224 met again in joint session to review / revise test assertions for P1224. Draft 4 of P1224 has already entered ballot and we agreed to assist them in ballot resolution as time permits. Test assertions will be balloted in a recirculation. Since P1224 is a normative reference for POSIX.17, a stable version is essential for our ballot.

We sent a representative to POSIX.0's Architecture Framework BOF where the results of their recent Mock Ballot were discussed. POSIX.17 had submitted comments/objections to the POSIX.0 Mock Ballot (Draft 14), focusing on the "networking" section. We were told that all our comments and objections were accepted and will be included in the next draft. The POSIX.0 model defined in the Mock Ballot draft seemed to recognize the need for APIs aimed at systems integrators as well as end users.

POSIX.17 shares a problem with P1224 and P1224.1. It seems that the objects defined in the base documents (XDS, XOM, X.400 API) reserved object ids (OIDs) in a vendor's (DEC) registered ISO name space. This might be ok for vendor consortia, but it won't cut it for a de jure standard. Because this issue touches more than one group, the DSSC discussed it and agreed to produce a recommendation on how to proceed by next meeting.

In Closing

Again, there are quite a few homework assignments between meetings. (I think there's a trend here.) Given this is our last quarter before ballot, we need to complete formation of the ballot group, fix the test assertions, finalize Draft 3, and respond formally to our Mock Ballot reviewers. We've also been asked by the DSSC and PMC to split our current Project Authorization Request (PAR) into two new PARs, one which addresses only the API to directory services and the other

which addresses the POSIX name space issue.

The group will reconvene April 6-10, 1992 at the IEEE POSIX meeting in Dallas.

Report on the POSIX Study Group on Distributed Security

Laura Micks <uunet!aixsm!micks> reports on the January 15, 1992 meeting in Irvine, CA:

A study group has formed to investigate the feasibility of a project request (PAR) for Distributed Security.

One of the major topics raised at the Distributed Services Steering Committee (DSSC) was the problem of Security in a Distributed environment. This issue is not addressed by the Security working group (POSIX.6), nor any of the working groups under the DSSC.

A meeting was scheduled for all interested parties to discuss future directions in this area. Approximately 20 people attended and the application was made to be approved as a Study Group. If approved, a Study Group can be funded (from a logistics point of view) to meet for several meetings without an official PAR in place. The group plans to meet for an entire week next meeting cycle.

Most of the attendees were from the Security and Systems Management groups. Several people attended for general interest. It took the group quite some time to get rolling. There seemed to be two camps: one that wanted to define a conceptual model, identify services required, etc., and the other that wanted to pin down the existing implementations, choose one, and tweak it where necessary.

A PAR was actually drafted in October 1991 by Data Logic on behalf of Petr Janeczek of X/Open. The PAR was not officially submitted to the POSIX Sponsor Executive Committee, probably due to potential lack of support and sponsorship within the POSIX community. The draft of this PAR was copied and distributed to the study group.

Known existing projects and organizations working on similar efforts were identified. The known models identified were as follows:

Open Software Foundation's Distributed Computing Environment (DCE)

NIS (Sun)

ECMA TC46 Technical Committee on Security Framework

ISO 7498-2 Security Addendum covering Architectural Framework/Security Svcs

The Andrew File System (AFS)

Project Athena

GSSAPI - A generic security API from DEC

Project MAXSIX

DNSIX - (Mitre)

Netware

GASSP (Generally Accepted Security System Principles)

U.S. Government OSI Profile (GOSIP)

We decided to further the study by arranging as many presentations as feasible from the list above for the April meeting. The meeting agenda will be to hear the architectural presentations on security models, and to determine selection requirements for base documents. A thorough evaluation will be made at the July meeting.

It is premature to assess the viability of this study group becoming an actual POSIX committee. The initial meeting was somewhat disorganized but in all fairness, there was little or no advance notice of this group's meeting, hence the attendees were unprepared. Given the sensitivity of the subject and the obvious differences of opinions raised at the January meeting, I don't expect that the exercise of selecting a particular model to be used as a base document will be trivial.

The Bookworm

Peter Salus, Sun User Group, Inc.
<peter@sug.org>

Objects

We're all into objects now, and it seems as though every publisher has discovered that what they thought was C++ or Eiffel or Smalltalk can be poured into a different marketing container. Though I admit that I liked Timothy Budd's *Introduction to Object-Oriented Programming* (Addison-Wesley, 1991) and Gorlen-Orlow-Plexico's *Data Abstraction and Object-Oriented Programming in C++* (John Wiley & Sons, 1990), which came with a tear-out card so you could get the NIH Class Libraries, which Keith Gorlen talked about at the 1991 USENIX C++ Advanced Topics Workshop, two new books have caught my eye.

The first is an out-and-out textbook: *Object-Oriented Languages, Systems and Applications*, edited by Gordon Blair, John Gallagher, David Hutchison, and Doug Shepherd (Halsted Press/John Wiley, 1991; 378pp.; \$34.95). This volume demonstrates just how far apart the UK and the USA are. There are fourteen chapters, ranging from introductory principles to 30 pages each on REKURSIV and BETA (the former from Linn Smart in Glasgow, the latter from Norway). I recognize that cultural chauvinism is rampant, but I believe that a good discussion of C++ or Eiffel would be of greater benefit to a student. The bibliographies, too, don't reflect much of what has appeared over the past few years: can it be that news of the USENIX C++ events has not gotten to the UK? Is it possible that the most recent work on Chorus is 1986, on Clouds is 1988, and on Guide is also 1988? For a book published earlier in 1991, Budd's seems more up-to-date.

The second volume is a true gem: David A. Taylor's *Object-Oriented Technology: A Manager's Guide* (Addison-Wesley, 1991; 147pp.). This was originally published in 1990 by the Servio Corporation, as an attempt at explaining the technology to managers, members of the sales force, etc. It's just great! It is clear, has lots of diagrams, has no code, and should be just right for a short plane trip or two. Probably every manager and marketer without technical background should be given a copy of Taylor's book. This volume will take tens of points off your blood pressure after trying to explain stuff to your management.

Driving license

Device drivers seem to be strange and arcane things: we all know just how important they are. Device drivers are basically translators, taking those generic requests from the OS and delivering comprehensible commands to peripheral controllers.

Hardly anyone, so far as I can tell, has mastered enough of the arcana to be able to just sit down and write code. I once looked at the notes for Dan Klein's device drivers tutorial and decided that it wasn't for me. I've now looked into George Pajari's *Writing UNIX Device Drivers* (Addison-Wesley, 1992; \$32.95) and decided that it still isn't my thing. But at the same time, I've learned a lot. And I'll tell you that Pajari's is a useful book. It has drawbacks, but so do all other books.

Pajari's plusses lie in the areas of organization and examples. The book appears well-organized and I was able to follow the line of reasoning. Better still, there are literally hundreds and hundreds of lines of code. The illustrative examples may well be the best part of the book. For another \$39.95, you can order a 3.5" or 5.25" floppy with sample device drivers from Pajari's company. The disk comes in either MS-DOS or SCO tar format. If you really need to write drivers, this may be a bargain: the book and the floppy come to less than half of what a tutorial would run you. And it is a decent book.

On the other hand, you can always copy a driver that works and fiddle with it for some new peripheral.

A Library book

I'll admit to being a fan of P. J. Plauger: I learned a lot from both *Software Tools* and *Elements of Programming Style*. Plauger has now produced *The Standard C Library* (Prentice-Hall, 1992; 498pp.). In some ways, this is an ANSI X3.159-1989/ISO/IEC 9899:1990 version of Section 3 of the Programmer's Reference. But each chapter (from <assert.h> to <time.h>) contains sections on "Background" and "What the C Standard Says." Plauger's chapters also contain full references. Just in case you were going to use this in class, there are Exercises at the end of each chapter.

Best of all, Plauger appears to have tested his code with the Gnu, Sun, Borland, CenterLine [=Saber], UNIX, VAX, and ULTRIX compilers. Though I admit to not having tested any code, it ought to compile and run.

Some history: In May 1989, I was given a 24-page paper, "Project Athena as a Distributed System,"

by George A. Champine of DEC. For five years, from 1986 to 1991, Champine served as DEC's Associate Director of Project Athena. Champine has now written *MIT Project Athena: A Model for Distributed Campus Computing* (Digital Press/Prentice Hall, 1991; 282pp.; \$29). Divided into four sections (Development, Pedagogy, Technology, and Administration), this book gave me both a feeling for the vision that led (in 1983) to the beginnings of Athena and an appreciation of its genuine achievements, of which the X windowing system may well be the best-known example. This is a fine book. It should be read carefully by anyone interested in distributed computing, not merely those concerned with academic campuses.

Coming Attraction

W. Richard Stevens, *Advanced Programming in the UNIX Environment* (Addison-Wesley, 1992; over 700 pages) is supposed to be available at the very end of May. In some ways, it is an expansion of

Sections 2 and 3 of the Programmer's Reference Manual ("System Calls" and "C Library Routines"). But it is actually a lot more. First of all, Stevens' introductory chapters discuss basics ("logging in," "error handling," and "UNIX time values" in Chapter 1; "UNIX Standardization" and several "Implementations" [mainly SVR4 and 4.3+] in Chapter 2) not found in the PRM. Secondly, Stevens has put in thousands of lines of code in the form of examples and a lot of text in the form of rationales. In view of the fact that Plauger spends 500 pages on the C Libraries, I guess the additional 200 on System Calls is quite economical. Stevens has organized his Library material differently from Plauger, but this won't deter the avid reader at all. This is a useful volume from a skilled author. It deserves a real review, but I wanted this short note to get in to alert everyone. Addison-Wesley hopes to have finished books available at USENIX in San Antonio (their sending me an early copy was not totally innocent).

Book Review

Practical UNIX Security

by Simson Garfinkel and Gene Spafford
O'Reilly & Associates, 1991
ISBN 0-937175-72-2

Reviewed by George W. Leach
<jc3b23!gwl@uunet.UU.NET>

Practical UNIX Security is an excellent book, jam-packed with practical information, yet easy to read and even entertaining. It covers both System V and Berkeley derived variants of UNIX with a major concentration on networked systems. The authors not only cover UNIX security, but also reveal a great deal about the inner workings of UNIX as well along the way. For the practitioner, who may not be very familiar with UNIX internals, this is a tremendous bonus.

The book is organized into five parts and a set of appendices that cover UNIX and UNIX security basics, enforcing security, communications security, how to handle security incidents, and other security topics.

Part One on UNIX and UNIX security basics will be quite familiar to the experienced UNIX user. Topics covered include user ids, passwords, file system permissions, etc.... Yet, viewed from the perspective of how individual users must take responsibility for ensuring security, these aspects of UNIX take on a new importance. Frequently, while reading this portion of the book, I would have to stop and shake my head concerning how lax we all have become regarding these responsibilities. Do you leave your terminal or workstation logged on and unattended for lengthy periods of time? How often do you see people writing passwords down on a white board or on a piece of paper attached to a workstation? In addition to providing the reader with newfound perspective on everyday security issues associated with UNIX usage, this material can easily make everyone feel at ease with the subject matter by providing a common reference point for discussing advanced security issues.

Part Two addresses what a system administrator can do to prevent or reduce the chances of a

breakin, and to limit the potential damage that can occur in the event of a security incident. The seasoned system administrator will probably not find much new here. However, obtaining knowledge in this area has traditionally been through word of mouth and experience. With the tremendous growth of UNIX and networked environments over the past several years, there is certainly a large audience of budding system administrators who very much need this kind of guidance. Topics covered include strengthening user accounts against breakins, securing your data (eg., backups), effective use of the log files, and protecting against programmed threats.

Part Three expands the discussion of security to the area of communications between UNIX machines over a network or between remote users and machines. The initial chapters of this section discuss the traditional communications found on all UNIX machines, namely modems, uucp, cu, and tip. From there the topic coverage moves on to TCP/IP networking facilities, administration and applications, including ftp, rlogin, telnet, X Windows. Two more chapters cover Sun's NFS, MIT's Kerberos, and Sun's Secure Remote Procedure Call. The final chapter of this section discusses setting up a firewall machine between your internal network and the outside world (however you define internal and outside). The only complaint here is the omission of coverage of AT&T's Remote File System (RFS). While not as important at the time the authors were writing this book, interoperability between UNIX and non-UNIX environments is becoming more commonplace in commercial environments, thus adding to the security burden of a system administrator. If there is a second edition, this might be fertile ground for additional coverage.

Part Four, while not dealing with technical aspects of security, is perhaps the most valuable part of the book. It deals with what action to take in response to the occurrence of a security incident. This information is critical to the system

administrator or manager in charge of such a situation. While events such as the Internet Worm may have heightened awareness of security problems, there are still many people who need guidance if they should find themselves faced with such a problem. The chapters deal with how to discover that there is a problem, what to do to alleviate the situation, and what legal recourse one can take or choose not to take in response to an incident.

Part Five deals with a couple of miscellaneous topics that are pertinent to all computer systems, not just UNIX. These are Encryption and Physical Security. There is some interesting history behind both of these topics as well as a great deal of current state of the practice application of the material to UNIX environments. Most books on security deal mostly with these two areas in a generic manner and rarely delve into the details of a specific operating system as this book does.

The appendices wrap up the book's coverage of security by providing a security checklist, a summary of important files on UNIX systems for security issues, how processes work in UNIX, how MIT's Kerberos authentication service works, and a reference section on other resources concerning security including printed references, descriptions and contact information for various organizations concerned with security, and available software resources.

This is one of the few technical books that has been able to hold my attention throughout its entirety. The writing style, organization, level of detail, and illustrations made reading an enjoyable experience. Security is an interesting and critical topic to open systems, yet many an author has successfully made it boring and laborious to read about. I would recommend this book to anyone involved with UNIX whether you are an administrator, programmer, user, or manager who is interested in learning about security as it pertains to UNIX and network environments.

Calendar of UNIX-Related Events

1992

- July 3 UKUUG, London, UK
13-17 IEEE 1003, Chicago, IL
20-22 Sun Open Sys. Expo, Anaheim, CA
27-31 Siggraph, Chicago, IL
- Aug 10-13 * C++, Portland, OR
18 DKUUG, Helsingor, Denmark
- Sep 8-11 AUUG, Melbourne, Australia
14-17 * UNIX Security III, Baltimore, MD
22-24 UNIX Expo, New York, NY
24 DKUUG, Copenhagen, Denmark
- Autumn ISO/IEC JTC1 SC22 WG15
Denmark
NUUG, Norway
SUUG, Soviet Union
- Oct 5 - 9 NLUUG, Amsterdam,
The Netherlands
6 WG15, Denmark
18-22 OOPSLA, Vancouver, Canada
19-23 * LISA VI, Long Beach, CA
19-23 IEEE 1003, Utrecht, The Netherlands
26-30 Interop, San Francisco, CA
29 DKUUG, Odense, Denmark
- Nov 25-27 EurOpen/UniForum
Utrecht, The Netherlands
26 DKUUG, Copenhagen, Denmark
- Dec 7 Sun User Group, San Jose, CA
UKUUG/UKnet, Manchester, UK

1993

- Jan 11-15 ISO/IEC JTC1 SC22 WG15,
New Orleans, LA
25-29 * USENIX, San Diego, CA
- Feb 22-24 Sun Open Sys. Expo, Chicago, IL
- Mar 15-18 UniForum, San Francisco, CA
24-31 CeBIT 93, Hannover, Germany
- Spring * Mach
* UNIX Applications Development
- Apr 19-23 ISO/IEC JTC1 SC22 WG15.
Irvine, CA

- May 3- 7 EurOpen, Seville, Spain
- Jun 21-25 * USENIX, Cincinnati, OH
- Jul 12-16 ISO/IEC JTC1 SC22 WG15
Autumn EuroOpen/UniForum
Utrecht, The Netherlands
* LISA VII, West Coast, USA
* SEDMS
- Oct 18-22 ISO/IEC JTC1 SC22 WG15
25-29 Interop, San Francisco, CA

1994

- Jan 17-21 * USENIX, San Francisco, CA
- Feb 14-17 UniForum, Dallas, TX
- Mar 16-23 CeBIT 94, Hannover, Germany
23-25 UniForum, San Francisco, CA
- Apr 18-22 EurOpen
- Jun 6-10 * USENIX, Boston, MA
- Sep 12-16 Interop, San Francisco, CA
- Autumn EuroOpen/UniForum
Utrecht, The Netherlands

1995

- Jan 16-20 * USENIX, New Orleans, LA
- Feb 21-23 UniForum, Dallas, TX
- May 1- 5 EurOpen
- Jun 19-22 * USENIX, San Francisco, CA

1996

- Jan 22-26 * USENIX, San Diego, CA
Mar 11-14 UniForum, San Francisco, CA

This is a combined calendar of planned conferences, workshops, and standards meetings related to the UNIX operating system. If you have a UNIX-related event that you wish to publicize, please contact login@usenix.org. Please provide your information in the same format as above. This calendar has been compiled with the assistance of Alain Williams of EurOpen.

* = events sponsored by the USENIX Association.

Call for Papers

USENIX Winter 1993 Technical Conference San Diego, California January 25-29, 1993

The Challenge of Innovation

UNIX and its cousins find themselves in increasing use throughout the industry. Succeeding in the challenge of meeting the world's expectations of software is an increasingly difficult task.

This USENIX conference is looking for innovative papers in a variety of areas, for example:

- Computing in the very large
 - Global connectivity
 - Coping with explosive growth
- Distributed environments
 - Client/server
 - Location independent computing
- Sociological and societal impacts
 - Connectivity vs. Security
 - New base applications
- Utilizing state-of-the-art hardware
 - Storage systems
 - Communications systems
 - Large networks
- Exploiting increasingly layered software
 - Object oriented systems
 - Creative new interfaces
 - Complexity management
 - New development techniques
- Visionary base systems software
 - New building blocks
 - Novel leverage of standards

At the USENIX Winter 1993 Technical Conference, systems researchers and developers, systems administrators, software professionals, programmers, applications developers, support staff, technical managers, and educators tackle questions of immediate importance to advanced computing systems development and management.

The Program Committee solicits new work on all topics related to UNIX or UNIX-inspired programming and technologies. Vendors are welcome to submit technical presentations, but the program committee will reject product announcements.

Relevant Dates for Refereed Paper Submissions

Extended Abstracts Due: **July 20, 1992**
Notifications to Authors: **August 19, 1992**
Final Papers Due: **November 20, 1992**

Form of Refereed Paper Submissions

Submissions must be in the form of extended abstracts (1500-2500 words or 3-5 pages in length). Shorter abstracts might not give the program committee enough information to judge your work fairly and, in most cases, your submission will be rejected. Longer abstracts and full papers simply cannot be read by the committee in the time available. Feel free to append a full paper to an extended abstract; this is sometimes useful during evaluation. The extended abstract should represent your paper in "short form." The committee wants to see that you have a real project, that you are familiar with the work in your area, and that you can clearly explain yourself.

A Good Extended Abstract Should Contain:

- Abstract (same as in the final paper);
- Introduction (to the problem & its importance);
- Solution (details on the problem and its issues, design decisions, tradeoffs, motivations, implementation details);
- Evaluation;
- References to previous work

Every Submission Should Include:

- The extended abstract;
- One contact author with a daytime phone number and surface mail address;
- Email address, if available;
- Home phone number (volunteers work at night!);
- Indication if any authors are students;
- List of audio/visual equipment desired beyond microphone and overhead projector.

Please note: presentations are usually scheduled for 25 minutes.

Where to Send Submissions and Make Inquiries

Six paper copies of each submission should be sent to:

Rob Kolstad
7759 Delmonico Drive
Colorado Springs, CO 80919

Make inquiries regarding submissions to:
Rob Kolstad at (719) 593-9445 or via email to kolstad@bsdi.com.

Tutorial Program

At San Diego, USENIX will offer tutorials such as:

Topics in Systems Administration;
Distributed File System Administration;
UNIX Programming Tools;
Systems and Network Security
Kernel Internals: OSF/1, SVR4, 4.4BSD;
Developing & Debugging X-Based Applications;
Network Program Maintenance and Design;
Introductions to C++ and Perl;
Micro-Kernel Technologies;
POSIX Threads and Systems Programming

In an effort to continue to provide the best possible tutorials, USENIX is soliciting proposals for new tutorials. If you are interested in presenting a tutorial, contact the Tutorial Coordinator:
Daniel V. Klein <dvk@usenix.org>.

Invited Talks

Invited Talks Coordinators:
Tom Cargill, Consultant
Bob Gray, US WEST Advanced Technologies
(303) 494-3239
<ITusenix@usenix.org> or
<uunet!usenix!ITusenix>

As part of the technical sessions, a full series of invited talks provide introductory and advanced information about a variety of interesting topics, such as using standard UNIX tools, tackling system administration difficulties, or employing specialized applications. We welcome suggestions for topics as well as request proposals for particular Talks. In your proposal, state the main focus, include a brief outline, and be sure to emphasize why your topic is of general interest to our community.

Conference Program Committee

General Chair:

Rob Kolstad, *Berkeley Software Design, Inc.*

Technical Program Chair:

Dan Geer, *Geer Zolot Associates*

Matthew Blaze, *Princeton University*

Tom Christiansen, *CONVEX Computer*

Clement T. Cole, *Locus Computing Corp.*

James Duncan, *Pennsylvania State University*

Dick Dunn, *eklektix*

Peter Honeyman, *CITI, University of Maryland*

Daniel V. Klein, *Software Engineering Institute*

Steve McDowell, *Exlog, Inc.*

Dinah McNutt, *Tivoli Systems, Inc.*

Kent Peacock, *Intel Corporation*

Gretchen Phillips, *State Univ. of New York /Buffalo*

David S. H. Rosenthal, *SunSoft, Inc.*

Jeffrey R. Schwab, *Purdue University*

Mary Seabrook, *Open Systems Solutions, Inc.*

Dave Taylor, *SunWorld Magazine*

Saul G. Wold, *Sun Microsystems*

Awards for Best Paper and Best Student Papers

A cash prize will be awarded by the conference program committee for both the best paper and the best paper by a full-time student at the conference. With your submission, please indicate if you are a full-time student.

For More Conference Information

Materials containing all details of the technical and tutorial program, conference registration, hotel and airline discount and reservation information will be mailed in September 1992.

Call for Papers

USENIX Systems Administration Conference (LISA VI) Long Beach, CA October 19-23, 1992

The annual LISA conference provides a forum in which system administrators from a variety of sites can meet to share new ideas and experiences. A growing success for the past five years, LISA is the only conference which focuses specifically on the needs of system administrators. In previous years, LISA has targeted large installations. However, this year we are extending the scope of LISA to include system administrators from all UNIX sites.

The Sixth USENIX System Administration Conference (LISA) will be held at the Long Beach Sheraton hotel in Long Beach, California, on October 19 - 23, 1992. A dual-track tutorial program will be offered during the first two days of the conference, followed by a three day technical conference. The tutorial program will address issues in introductory and advanced system administration.

The program committee will be reviewing papers submitted on subjects including (but not limited to):

- Tools for Real-Time System Troubleshooting
- Remote/Off-site System Administration
- Tricks in User Education
- Graphical User Interfaces for System Administration
- Distributed System Administration
- Experiences Using Third-party Administration Software
- Network Growth and Performance Management
- How to Grow Your Own Junior System Administrators
- Network Management
- Wireless LANs
- System Security Monitoring
- Evaluating Performance of High-End Workstations and Servers
- Keys to Successful, Painless Upgrades
- Object Management Systems for System Administration.
- Standardization of System Administration

- Heterogeneous System Administration
- System Archiving and Backups

We are especially interested in papers which provide freely available or fully described solutions to existing problems. We are also looking for papers which, in some way, advance the state of the art.

The committee requires that an extended abstract be submitted for the paper selection process [full-papers are not acceptable for this stage; if you send a full paper, you must also include an extended abstract for judging]. Your extended abstract should consist of a traditional abstract which summarizes the content/ideas of the entire paper, followed by a skeletal outline of the full paper. Final papers should be from 5 to 20 pages in length, including diagrams and figures. Papers should include a brief description of the site, an outline of the problem and issues, and a description of the solution. We *require* electronic form of the extended abstract; we require both hardcopy and electronic (nroff/troff or ASCII) form of the final paper.

Conference proceedings will be distributed to all the attendees and also will be available after the conference from the USENIX Association.

In addition to tutorials and regular technical sessions, a handful of other events will be included as part of the program. For example, the program may include special panels, work-in-progress reports, birds-of-a-feather (BOF) sessions, and invited talks. The program committee invites you to submit informal proposals, ideas, or suggestions you might have on any of these topics.

Important Dates

Extended Abstract Deadline: **June 29, 1992**
Acceptance Notification: **July 20, 1992**
Final Papers Received: **August 31, 1992**

Contact Information

Submit electronic copy of extended abstracts (preferably by electronic mail) to:

Trent Hein
XOR Computer Systems
2525 Arapahoe, Suite E4-264
Boulder, Colorado 80302
(303) 440-6093
trent@xor.com

Keynote Speaker
Doug Kingston, *Morgan Stanley & Co.*

Alternate Track Coordinator
Steve Simmons, *Industrial Technology Institute*

Terminal Room Coordinator
Barb Dyker, *University of Colorado, Boulder*

BOF Coordinator
Arch Mott, *Protocol Engines, Inc.*

Vendor Display Coordinator
John Donnelly, *Seminars, Meetings, and Exhibits,
A Planning & Management Company*

Program Committee

Trent Hein, *XOR Computer Systems*
Rik Farrow, *UNIX World*
Jeff Forys, *University of Utah*
John Hardt, *Martin Marieta Astronauts*
Rob Kolstad, *Berkeley Software Design, Inc.*
Herb Morreale, *XOR Computer Systems*
Pat Parseghian, *AT&T Bell Laboratories*
Jeff Polk, *Berkeley Software Design, Inc.*

C++ Conference

Portland, Oregon
August 10-11, 1992

Tutorials

Monday, August 10

C++ Programming Style
Tom Cargill, Consultant

Intended Audience:

This tutorial will be of value to programmers who are starting to program in C++, or have a reading knowledge of the language, and are looking for guidance on how to use its features in practice. Knowledge of C++ language basics is assumed; if need be, language features will be clarified briefly. The tutorial material is code intensive, for programmers who like to read and understand programs.

Course Description:

C++ supports programming-in-the-large, allowing relationships between different parts of a program to be described. The scope of C++ programming style therefore goes beyond the issues of traditional programming-in-the-small, such as indention and the use of goto. This tutorial examines the use of language features that often confuse even experienced programmers. Unwarranted use of the more powerful features leads to cluttered programs that are harder

to comprehend, and in some cases less efficient, than more straightforward alternatives.

In this tutorial we examine, and then simplify, a number of programs. The techniques range from simple rules of thumb about constructors to transformations that remove redundant inheritance. We read programs, discuss their organization and use of C++, critique their design, redesign where necessary, and then recode. The discussion ranges from questions of data abstraction and object-oriented design to the expression of a given design in C++. Design and coding style guidelines are distilled from the examples.

Using OOD with C++
Michael J. Vilot, ObjectWare, Inc.

Intended Audience:

Intermediate to advanced C++ developers, preferably with several months experience using C++ to build real-world applications.

Course Description:

This tutorial presents an object-oriented design method and illustrates ways to implement object-oriented designs using C++. The method is described in detail in the book "Object-Oriented Design with Applications" by Grady Booch (Benjamin-Cummings, 1990).

A key focus for this tutorial is how certain features of the C++ language support implementation of object-oriented designs.

Although C++ does not have to be used as an object-oriented programming language, many of its important features directly support this approach. Those attending the tutorial should gain some insights on effectively using these features to realize their designs.

The tutorial will cover the following topics:

- Fundamental Concepts of C++ and OOD
- C++ Support for OOD
- Adapting OOD to C++
- Development Process and Pragmatics

Tuesday, August 11

Designing and Implementing Effective Classes
Scott Meyers

Intended Audience:

Programmers and managers involved in the design and implementation of C++ classes for real products. Participants should already know C++, but expertise is not required. People who learned C++ through an earlier tutorial, as well as people who have been programming in C++ for some time, should come away from this tutorial with useful practical information.

Course Description:

This tutorial emphasizes that creating new classes in C++ is creating new {types}, focuses on the subtleties involved in specifying and implementing effective classes. For class interfaces, the primary issues to be addressed include usability, compatibility with primitive types, completeness, maintainability, and extensibility. For class implementations, the main issues are correctness, maintainability, and efficiency.

Topics covered include:

- Class interface design: public vs. private members, virtual vs. nonvirtual functions, minimalness and completeness, avoiding name conflicts, etc.
- Declaration and implementation of critical functions: constructors, destructors, assignment operators, input and output operators.
- Overloading operators: when, how, and where; gotchas.
- The different meanings of {const}; how and when to use them.
- Making appropriate design decisions --

choosing between:

- member functions, global functions, and friend functions.
- virtual and nonvirtual functions.
- default parameters and overloaded functions.
- public inheritance and templates.
- private inheritance and layering.

After completing this tutorial, participants will be aware of many of the often-surprising rules of thumb that expert designers and programmers routinely apply to their work in C++.

Designing and Coding Reusable C++, *Martin Carroll and Margaret A. Ellis, AT&T Bell Labs*

Intended Audience:

Programmers with a working knowledge of C++ who want to learn what it takes to write truly reusable C++ code that will satisfy their intended customers needs.

Course Description:

Writing reusable code is much harder than writing code intended for use in a single program. This will show, through a wealth of program examples, what can go wrong when a C++ programmer tries to reuse apparently well-designed, well-implemented code. We will also present ways of preventing these problems in newly developed code.

The tutorial begins with an example showing the difference between usability and reusability. We then attempt to enumerate the thousand-and-one properties that have, at one time or another, been demanded of reusable code, and illustrate typical tradeoffs among these properties. We then divide the properties into two general categories, "properties of efficiency" and "properties of flexibility." We show ways to implement code so that it falls into one or the other of these categories. Next, we illustrate some pitfalls to be avoided when designing the interface of reusable code, and we discuss the relationship of the implementation to that interface. A major topic in reusable code is compatibility; we define the different kinds of compatibility, and then we illustrate ways of providing as much compatibility as possible.

Other topics will include:

- Potential versus actual program errors
- Efficient parameterized types
- Exceptions
- Shared libraries

- The static initialization problem
- Source code organization
- Portability

Each of these topics will be discussed in terms of what they mean for writers of reusable code. We will even have something concrete to say about documentation.

Technical Sessions

Wednesday, August 12

9:00 - 10:00 a.m. Keynote Address

The Essentials of Object-Oriented Programming, *Kristen Nygaard, Department of Informatics, University of Oslo*

Session 1: 10:30 - 12:30

Chair: **Doug Lea, SUNY Oswego**

Smart pointers: They're smart, but they're not pointers, *Daniel R. Edelson, INRIA Project SOR*

Not a language extension, *Martin D. Carroll, AT&T Bell Laboratories*

Garbage collection and run-time typing as a C++ library, *David Detlefs, Digital Equipment Corporation*

Encapsulating a C++ library, *Mark Linton, Silicon Graphics, Inc.*

Session 2 2:00 - 3:30

Chair: **Jim Waldo, SUN**

Sniff: A pragmatic approach to a C++ programming environment, *Walter R. Bischofberger, Union Bank of Switzerland*

A statically typed abstract representation for C++ Programs, *Robert B. Murray, AT&T Bell Labs*

CCEL: A metalanguage for C++, *Carolyn K. Duby, Scott Meyers, Steven P. Reiss, Brown University*

Session 3: 4:00 - 5:30

Chair: **Theodore Goldstein, SUN**

Space-efficient trees in C++, *Andrew Koenig, AT&T Bell Laboratories*

High-performance scientific computing using C++, *K. G. Budge, J. S. Perry, A. C. Robinson, Sandia National Laboratories*

O-R gateway: A system for connecting C++ application programs and relational databases, *Abdulah Alashqur, Craig Thompson, Texas Instruments*

Vendor Demos/Display 7:00 - 10:00 p.m.

Thursday, August 13

Session 4: 9:00 - 10:30 a.m.

Chair: **Keith Gorlen, NIH**

Static initializers: Reducing the value added tax on programs, *John F. Reiser, Mentor Graphics Corp.*

Cdiff: A syntax directed diff for C++ programs, *Judith E. Grass, AT&T Bell Laboratories*

C++ in a changing environment, *Andrew J. Palay, Silicon Graphics Computer Systems*

Session 5: 11:00 - 12:30

Chair: **Dag Bruck, Lund Institute**

Adding concurrency to a programming language, *Peter A. Buhr, Glen Ditchfield, University of Waterloo*

A portable implementation of C++ exception handling, *Don Cameron, Paul Faust, Dmitry Lenkov, Michay Mehta, Hewlett-Packard California Language Laboratory*

An assertion mechanism based on exceptions, *Philippe Gautron, Universite Paris VI, LITP-IBP*

Session 6: 2:00 - 3:30

Chair: **Susan E. Waggoner, US WEST**

A communication facility for distributed object-oriented applications, *Afshin Daghi, Pierre Delisle, Salil Deshpande, Sun Microsystems Inc.*

Writing a client-server application in C++ *Paulo Guedes, Open Software Foundation*

Integrating the Sun Microsystems XDR/RPC protocols into the C++ stream model, *Robert E. Minnear, Patrick A. Muckelbauer, Vincent F. Russo, Purdue University*

Session 7: 4:00 - 5:30 Run Time Type Identification

Chair: **Mark Linton, Silicon Graphics**

Run time type identification for C++ *Bjarne Stroustrup, AT&T Bell Laboratories, Dmitry Lenkov, Hewlett-Packard*

Panel Discussion: *Mark Linton, Silicon Graphics, others to be determined*

Friday, August 14: Advanced Topics Workshop

The focus of this year's workshop will be representations of C++ programs as the basis for tools for C++ software development: what information should be included in the internal representation, building the representation using full and fuzzy parsers, and using and modifying the representation in programming tools.

This workshop will provide a forum for representation developers to explain their design decisions and for tool developers to describe what an ideal representation for C++ should offer. Admission is by invitation only. Anyone wishing an invitation must submit a one or two page position

paper describing their interest in the topic of the workshop. Authors of papers submitted to the conference are invited automatically.

Security Symposium

Baltimore, MD
September 14-16, 1992

Tutorial Program
Monday, September 14, 1992

Network Security: The Kerberos Approach

Dan Geer, Geer/Zolot Associates and Jon A. Rochlis, MIT

Intended Audience: Systems developers responsible for networked workstation environments, particularly those whose environments may include networks which are not themselves physically secure (i.e., "open" networks) and systems managers concerned about the inherent lack of security for managing today's network-based environments (e.g., UNIX's *.rhosts* files).

The amazing and constantly growing numbers of machines and users ensures that untrustworthy individuals have full access to the Internet. Given the increasing importance of the information transmitted, it is imperative to consider the basic security issues present as large open networks replace isolated timesharing systems.

This tutorial will focus on the challenges of providing security for cooperative work arrangements consistent with the location and scale independence of today's open networking environment. Attendees will gain an understanding of the kinds of security threats which result from operating in an open environment, such as one composed of a network of workstations and supporting servers. Effective approaches to meeting these threats will be presented. Although emphasis will be on the Kerberos system developed at MIT, public key techniques for ensuring privacy and authentication on an open network will be explored. The X.509 authentication model and the new Internet Privacy Enhanced Electronic Mail RFC's will be discussed.

Internet System Administrator's Tutorial

Ed DeHart and Barb Fraser, Computer Emergency Response Team

Intended Audience: This tutorial is designed for users and system administrators of UNIX systems. It is especially suited for system administrators of UNIX systems connected to a wide area network based on TCP/IP such as the Internet. Some system administrator experience is assumed.

The information presented in this tutorial is based on incidents reported to the Computer Emergency Response Team. The topics covered include:

System administration – defensive strategies

- Password selection
- Default login shell for unused accounts
- Network daemon configuration
- Verification of system programs
- System configuration files
- Searching for hidden intruder files
- Staying current with software releases
- Standard accounting files
- NFS configuration

System administration – offensive strategies

- COPS
- /bin/passwd replacement programs
- TCP/IP packet filtering
- TCP/IP daemon wrapper programs
- Security in programming

Site-specific security policies

- Maintaining good security at your site
- Providing guidance to users
- Handling incidents in an effective orderly fashion
- Reviewing Site Security Policy Handbook (RFC 1244)

Incident handling

- What to do if your site is broken into?

The technical program for Tuesday and Wednesday, September 15-16 will be set in June. Materials containing the details of the technical and tutorial program, registration and hotel reservation information will be mailed to the membership in July, 1992.

ORDER FORM
CONFERENCE & WORKSHOP PROCEEDINGS

Qty	Proceedings		Member Price	Non-Member Price	Subtotal	Overseas Postage	Total
WINTER/SUMMER CONFERENCES							
___	San Antonio	Summer '92	\$23	\$30	\$ _____	\$14	\$ _____
___	San Francisco	Winter '92	30	39	\$ _____	22	\$ _____
___	Nashville	Summer '91	32	38	\$ _____	22	\$ _____
___	Dallas	Winter '91	28	34	\$ _____	18	\$ _____
___	Anaheim	Summer '90	22	22	\$ _____	15	\$ _____
___	Washington, DC	Winter '90	25	25	\$ _____	15	\$ _____
___	Baltimore	Summer '89	20	20	\$ _____	15	\$ _____
___	San Diego	Winter '89	30	30	\$ _____	20	\$ _____
___	San Francisco	Summer '88	29	29	\$ _____	20	\$ _____
___	Dallas	Winter '88	26	26	\$ _____	15	\$ _____
___	Phoenix	Summer '87	35	35	\$ _____	20	\$ _____
___	Washington, DC	Winter '87	10	10	\$ _____	15	\$ _____
___	Atlanta	Summer '86	37	37	\$ _____	20	\$ _____
___	Denver	Winter '86	25	25	\$ _____	15	\$ _____
___	Portland	Summer '85	45	45	\$ _____	25	\$ _____
___	Dallas	Winter '85	15	15	\$ _____	10	\$ _____
___	Salt Lake City	Summer '84	29	29	\$ _____	20	\$ _____
___	Washington, DC	Winter '84	25	25	\$ _____	15	\$ _____
___	Toronto	Summer '83	32	32	\$ _____	20	\$ _____
___	San Diego	Winter '83	28	28	\$ _____	15	\$ _____
LARGE INSTALLATION SYSTEMS ADMINISTRATION							
___	LISA V	Sept. '91	20	23	\$ _____	11	\$ _____
___	LISA IV	Oct. '90	15	18	\$ _____	8	\$ _____
___	LISA III	Sept. '89	13	13	\$ _____	9	\$ _____
___	LISA II	Nov. '88	8	8	\$ _____	5	\$ _____
___	LISA I	April '87	4	4	\$ _____	5	\$ _____
C++							
___	C++ Conference	Apr. '91	22	26	\$ _____	11	\$ _____
___	C++ Conference	Apr. '90	28	28	\$ _____	18	\$ _____
___	C++ Conference	Oct. '88	30	30	\$ _____	20	\$ _____
___	C++ Workshop	Nov. '87	30	30	\$ _____	20	\$ _____
SECURITY							
___	UNIX Security II	Aug. '90	13	16	\$ _____	8	\$ _____
___	UNIX Security	Aug. '88	7	7	\$ _____	5	\$ _____
MACH							
___	Mach Symposium	Nov. '91	24	28	\$ _____	14	\$ _____
___	Mach Workshop	Oct. '90	17	20	\$ _____	9	\$ _____

Continued - see reverse

Qty	Proceedings		Member Price	Non-Member* Price	Subtotal	Overseas Postage	Total
DISTRIBUTED & MULTIPROCESSOR SYSTEMS (SEDMS)							
___	SEDMS III	Mar. '92	30	36	\$ _____	20	\$ _____
___	SEDMS II	Mar. '91	30	36	\$ _____	20	\$ _____
___	SEDMS	Oct. '89	30	30	\$ _____	20	\$ _____
GRAPHICS							
___	Graphics Workshop V	Nov. '89	18	18	\$ _____	10	\$ _____
___	Graphics IV	Oct. '87	10	10	\$ _____	10	\$ _____
___	Graphics III	Nov. '86	10	10	\$ _____	5	\$ _____
___	Graphics II	Dec. '85	7	7	\$ _____	5	\$ _____
OTHER WORKSHOPS							
___	File Systems	May '92	15	20	\$ _____	9	\$ _____
___	Micro-Kernel & Other Kernel Arch.	April '92	30	39	\$ _____	20	\$ _____
___	UNIX Transaction Processing	May '89	12	12	\$ _____	8	\$ _____
___	Software Management	Apr. '89	20	20	\$ _____	15	\$ _____
___	UNIX & Supercomputers	Sept. '88	20	20	\$ _____	10	\$ _____

Discounts are available for bulk orders. Please inquire.

Total price of Proceedings _____ **
Calif. residents add sales tax _____
Total overseas postage _____
Total enclosed _____

**If you are paying member price, please include member's name and/or
 membership number _____

PAYMENT OPTIONS*

___ Check enclosed - payable to USENIX Association

___ Charge my: ___ VISA  ___ MC  Account # _____ Exp. Date _____

___ Purchase order enclosed Signature _____

* Outside the USA? Please make your payment in US currency by one of the following:

- Check - issued by a local branch of a US Bank
- Charge (VISA, MasterCard, or foreign equivalent)
- International postal money order

Shipping Information

Ship to: _____

Please allow 2 - 3 weeks for delivery. Overseas orders are shipped via air printed matter.

Please mail or fax this order form with payment to:

USENIX Association
 2560 Ninth Street, Suite 215
 Berkeley, CA 94710
 FAX 510/548-5738

• If you are not a member and wish to receive our membership information packet, please check this box. ☐

SUBSCRIPTION ORDER FORM

PROCEEDINGS

Please enter my one-year subscription* to the 1992 USENIX Conference, Workshop and Symposia Proceedings, which include:

San Francisco Technical Conference - January 1992

Symposium on Experiences with Distributed & Multiprocessor Systems (SEDMS) III - March 1992

Workshop on Micro-Kernels and other Kernel Architectures - April 1992

File Systems Workshop - May 1992

San Antonio Technical Conference - June 1992

C++ Conference - August 1992

UNIX Security III Symposium - September 1992

Systems Administration (LISA) VI Conference - October 1992

***NOTE:**

Corporate, Educational and Supporting Members of USENIX automatically receive a subscription to all proceedings published during their term of membership, which may overlap with this offer.

COST: USENIX members: \$170.00

Non-members: \$214.00

Outside U.S.A. and Canada?

Add \$30.00 for surface postage.

Subscription Cost _____

**

Calif. Sales Tax _____

Overseas Postage _____

TOTAL ENCLOSED _____

** If you are paying member price, please include member's name and/or membership number _____

PAYMENT OPTIONS

☐ Check enclosed payable to USENIX Association. ☐ Purchase order enclosed.

☐ Please charge my: ☐ Visa ☐ MasterCard



Account # _____ Exp. Date _____

Signature _____

Outside the U.S.A.? Please make your payment in U.S. currency by one of the following:

- * Check - issued by a local branch of a U.S. Bank
- * Charge (Visa, MasterCard, or foreign equivalent)
- * International postal money order

4/92

Name _____

Address _____

City _____ State/Country _____ Zip/Postal Code _____

Please mail this order form to: USENIX Association
2560 Ninth Street, Suite 215
Berkeley, CA 94710
510/528-8649
FAX 510/548-5738
office@usenix.org

This offer is good through
December 31, 1992

The proceedings are shipped the week following the event.

Local User Groups

The Association will support local user groups by doing a mailing to assist in the formation of a new group and publishing information on local groups in *login*. At least one member of the group must be a current member of the Association. Send additions and corrections to:
login@usenix.org.

CA - Fresno:

The Central California UNIX Users Group consists of a uucp-based electronic mailing list to which members may post questions or information. For connection information:

Educational and governmental institutions:
Brent Auernheimer (209) 278-4636
brent@CSUFresno.edu or *csufres!brent*

Commercial institutions or individuals:
Gordon Crumal (209) 251-2648
csufres!gordon

CA - Irvine:

Meets the 2nd Monday of each month

UNIX Users Association of Southern California
Paul Muldoon (714) 556-1220 ext. 137
Horizons
Santa Ana, CA 92705

CO - Boulder:

Meets monthly at different sites. For meeting schedule, send email to *fruug-info@fruug.org*.

Front Range UNIX Users Group
Software Design & Analysis, Inc.
1113 Spruce St., Ste. 500
Boulder, CO 80302
Steve Gaede (303) 444-9100
gaede@fruug.org

FL - Coral Springs

S. Shaw McQuinn (305) 344-8686
8557 W. Sample Road
Coral Springs, FL 33065

FL - Western:

Meets the 1st Thursday of each month.

Florida West Coast UNIX Users Group
Richard Martino (813) 536-1776
Tony Becker (813) 799-1836
mcrsys!tony
Ed Gallizzi, Ph.D. (813) 864-8272
e.gallizzi@compmail.com
Jay Ts (813) 979-9169
uunet!pdn!tscs!metran!jan
Dave Lewis (407) 242-4372
dhl@ccd.harris.com

FL - Orlando:

Meets the 3rd Thursday of each month.

Central Florida UNIX Users Group
Mike Geldner (407) 862-0949
codas!sunfla!mike
Ben Goldfarb (407) 275-2790
goldfarb@hcx9.ucf.edu
Mikel Manitus (407) 869-2462
{codas,attmail}!mikel

KS or MO - Kansas:

Meets on 2nd Monday of each month.

Kansas City UNIX Users Group (KUUG)
813B St.
Blue Springs, MO 64015
(816) 235 5212
mig@cstp.umkc.edu

GA - Atlanta:

Meets on the 1st Monday of each month in White Hall, Emory University.

Atlanta UNIX Users Group
P.O. Box 12241
Atlanta, GA 30355-2241
Mark Landry (404) 365-8108

MI - Detroit/Ann Arbor:

Meets on the 2nd Thursday of each month in Ann Arbor.

Southeastern Michigan Sun Local Users Group
and Nameless UNIX Users Group
Steve Simmons office: (313) 769-4086
home: (313) 426-8981
scs@lokkur.dexter.mi.us
K. Richard McGill
rich@sendai.ann-arbor.mi.us
Bill Bulley
web@applga.uucp

MN - Minneapolis/St. Paul:

Meets the 1st Wednesday of each month.

UNIX Users of Minnesota
17130 Jordan Court
Lakeville, MN 55044
Robert A. Monio (612) 220-2427
pnessutt@dmshq.mn.org

MO - St. Louis:

St. Louis UNIX Users Group
P.O. Box 2182
St. Louis, MO 63158
Terry Linhardt (314) 772-4762
uunet!jgaltstl!terry

NE - Omaha:

Meets monthly.

/usr/group/nebraska
P.O. Box 31012
Omaha, NE 68132
Phillip Allendorfer (402) 423-1400

New England - Northern:

Meets monthly at different sites.

Peter Schmitt 603) 646-2085
Kiewit Computation Center
Dartmouth College
Hanover, HN 03755
Peter.Schmitt@dartvax!dartmouth.edu

NJ - Princeton:

Meets monthly.

Princeton UNIX Users Group
Mercer County Community College
1200 Old Trenton Road
Trenton, NJ 08690
Peter J. Holsberg (609) 586-4800
mccc!pjh

NY - New York City:

Meets every other month in Manhattan.

Unigroup of New York City
G.P.O. Box 1931
New York, NY 10116
Peter Gutmann (212) 618-0973
peterg@murphy.com

OK - Tulsa:

Meets 2nd Wednesday of each month.

Tulsa UNIX Users Group, \$USR
Stan Mason (918) 560-5329
tulsix!smason@drd.com
Mark Lawrence (918) 743-3013
mark@drd.com

TX - Austin:

Meets 3rd Thursday of each month.

Capital Area Central Texas UNIX Society
P.O. Box 9786
Austin, TX 78766-9786
officers@cactus.org
James Johnson (512) 331-3781
president@cactus.org

TX - Dallas/Fort Worth:

Dallas/Fort Worth UNIX Users Group
660 Preston Forest, Suite 177
Dallas, TX 75230
Kevin Coyle (214) 991-5512
kevincd@shared.com

TX - Houston:

Meets 3rd Tuesday of each month.

Houston UNIX Users Group
(Hounix) answering machine (713) 684-6590
Bob Marcum, President (713) 270-8124
Chuck Bentley, Vice-president
(713) 789-8928 *chuckb@hounix.uucp*

WA - Seattle:

Meets monthly.

Seattle UNIX Group Membership Info.
Bill Campbell (206) 947-5591
6641 East Mercer
Mercer Island, WA 98040-0820
bill@celestial.com

Washington, D.C.:

Meets 1st Tuesday of each month.

Washington Area UNIX Users Group
9811 Mallard Drive
Laurel, MD 20708
Alan Fedder (301) 953-3626

CANADA - Toronto:

143 Baronwood Court
Brampton, Ont. Canada L6V 3H8
Evan Leibovitch (416) 452-0504
evan@telly.on.ca

USENIX Association
2560 Ninth Street, Suite 215
Berkeley, CA 94710

SECOND CLASS
APPLICATION PENDING
at Berkeley, CA and
additional offices

POSTMASTER: Send address changes to *login.*, USENIX Association, 2560 Ninth Street, Suite 215, Berkeley, CA 94710

What's Inside?

What's Out There, Vol. 1:1

Can UNIX Designers Learn Anything from PC's?

Standards Activity Update

Election Results

Book Reviews